aws

Guía para desarrolladores

Amazon OpenSearch Service



Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon OpenSearch Service: Guía para desarrolladores

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Las marcas comerciales y la imagen comercial de Amazon no se pueden utilizar en relación con ningún producto o servicio que no sea de Amazon de ninguna manera que pueda causar confusión entre los clientes y que menosprecie o desacredite a Amazon. Todas las demás marcas registradas que no son propiedad de Amazon son propiedad de sus respectivos propietarios, que pueden o no estar afiliados, conectados o patrocinados por Amazon.

Table of Contents

¿Qué es Amazon OpenSearch Service? 1
Características de Amazon OpenSearch Service 2
Cuándo se debe usar 3
Versiones compatibles de Elasticsearch y OpenSearch 4
Soporte extendido y estándar 4
Versiones estándar compatibles y ampliadas 7
Cálculo de los cargos por soporte extendido 8
Precios 9
Servicios relacionados 9
Configuración 12
Concesión de permisos 12
Concesión de acceso programático 12
Configuración de la AWS CLI 14
Abra la consola 15
Introducción 16
Crear un dominio 17
Cargar datos para indexar 18
Opción 1: cargar un solo documento 19
Opción 2: cargar varios documentos 19
Buscar documentos 20
Para buscar documentos desde la línea de comandos 20
Buscar documentos mediante OpenSearch Dashboards 21
Eliminar un dominio 22
Amazon OpenSearch Ingestion 23
Conceptos clave 23
Ventajas 25
Limitaciones 25
Versiones de Data Prepper admitidas 26
Escala de canalizaciones 27
Precios 29
Regiones de AWS compatible 29
Configuración de roles y usuarios 29
Rol de canalización 31
Rol de ingesta 34

Concesión de acceso a dominios para canalizaciones 36
Concesión de acceso a colecciones para canalizaciones 40
Introducción a OpenSearch Ingestion 46
Tutorial: incorporar datos a un dominio 46
Tutorial: incorporar datos en una colección 55
Características de la canalización 62
Almacenamiento en búfer persistente 62
División 64
Encadenar 66
Colas de mensajes fallidos 67
Administración de índices 68
Reconocimiento integral 72
Contrapresión del origen 73
Creación de canalizaciones 74
Requisitos previos y rol de IAM requerido 74
Permisos de IAM necesarios 75
Especificar la versión de la canalización 77
Especificación de la ruta de ingesta 78
Creación de canalizaciones 78
Seguimiento del estado de creación de la canalización 83
Uso de esquemas 85
Visualización de canalizaciones 86
Actualización de las canalizaciones 89
Consideraciones 90
Permisos necesarios 90
Actualización de las canalizaciones 91
Implementaciones azul/verde para actualizaciones de canalización 92
Administración de los costos de las canalizaciones 93
Detención de una canalización 94
Inicio de una canalización 95
Eliminación de canalizaciones 96
Complementos y opciones compatibles 97
Complementos compatibles 97
Procesadores sin estado frente a procesadores con estado 100
Requisitos y restricciones de configuración 100
Integración de canalizaciones 105

Creación del punto de conexión de ingesta 106
Creación de un rol de ingesta 107
Servicios de Atlassian 109
Amazon Aurora 126
Amazon DynamoDB 156
Amazon DocumentDB 173
Confluent Cloud Kafka 192
Amazon MSK 202
Amazon RDS 210
Amazon S3 239
Amazon Security Lake 249
Fluent Bit 254
Fluentd 256
Inferencia en lotes de ML sin conexión 258
Recolector de OpenTelemetry 275
Kafka autogestionado 277
Clústeres de OpenSearch autoadministrados 284
Amazon Kinesis Data Streams 292
Pasos a seguir a continuación 305
AWS Lambda 305
Migración de datos entre dominios y colecciones 309
Limitaciones 309
OpenSearch Service como origen 310
Especificación de varios receptores de dominios de OpenSearch Service 312
Migración de datos a una colección de VPC de OpenSearch sin servidor 313
Administración de canalizaciones con los SDK de AWS 314
Python 314
Seguridad en OpenSearch Ingestion 318
Configuración del acceso a la VPC para canalizaciones 319
Configuración de canalizaciones para la ingesta entre cuentas 324
Identity and Access Management 332
Monitoreo con CloudTrail 340
Etiquetado de canalizaciones 344
Permisos necesarios 345
Uso de etiquetas (consola) 345
Uso de etiquetas (AWS CLI) 345

Registro y supervisión ... 346
Monitoreo de registros de canalización ... 346
Monitoreo de métricas de canalización ... 348
Prácticas recomendadas ... 381
Prácticas recomendadas generales ... 382
Alarmas de CloudWatch recomendadas ... 382
Amazon OpenSearch sin servidor ... 389
Ventajas ... 389
¿Qué es Amazon OpenSearch sin servidor? ... 390
Casos de uso de OpenSearch sin servidor ... 391
Funcionamiento ... 391
Elección de un tipo de colección ... 393
Precios ... 394
Regiones de AWS compatible ... 395
Limitaciones ... 395
Comparación entre OpenSearch Service y OpenSearch sin servidor ... 396
Tutorial: introducción a OpenSearch sin servidor ... 400
Paso 1: configurar permisos ... 401
Paso 2: crear una colección ... 402
Paso 3: cargar y buscar datos ... 403
Gestión de errores ... 404
Paso 4: eliminar la colección ... 405
Pasos a seguir a continuación ... 405
Creación y administración de colecciones ... 406
Administración de colecciones ... 406
Trabajo con colecciones de búsqueda vectorial ... 425
Uso de políticas de ciclo de vida de los datos ... 434
Administrar colecciones con los AWS SDK ... 443
Creación de colecciones con CloudFormation ... 454
Hacer copias de seguridad de las colecciones mediante instantáneas ... 456
Administración de los límites de capacidad ... 463
Como establecer los parámetros de capacidad ... 465
Límites de la capacidad máxima ... 466
Monitoreo del uso de la capacidad ... 466
Ingesta de datos en las colecciones ... 466
Permisos mínimos necesarios ... 467

OpenSearch Ingestion ... 468
Fluent Bit ... 468
Amazon Data Firehose ... 469
Go ... 470
Java ... 472
JavaScript ... 474
Logstash ... 476
Python ... 479
Ruby ... 480
Otros clientes ... 481
Configurar Machine Learning ... 482
Machine Learning ... 482
Connectors ... 483
Modelos ... 483
Configurar permisos para Machine Learning ... 484
API y características no compatibles ... 485
Configure la búsqueda neuronal y la búsqueda híbrida ... 486
Búsqueda neuronal ... 486
Búsqueda híbrida ... 486
Consultas neuronales e híbridas ... 487
Configuración de permisos ... 489
Configure los flujos de trabajo ... 491
Flujos de trabajo ... 491
Configuración de permisos ... 491
Seguridad en OpenSearch sin servidor ... 493
Políticas de cifrado ... 495
Políticas de red ... 495
Políticas de acceso a datos ... 496
Autenticación SAML e IAM ... 497
Seguridad de la infraestructura ... 498
Introducción a la seguridad ... 498
Identity and Access Management ... 513
Cifrado ... 531
Acceso a la red ... 542
Conformidad con FIPS ... 553
Control de acceso a los datos ... 559

Puntos de conexión de VPC ... 570
Autenticación SAML ... 582
Validación de conformidad ... 592
Etiquetado de colecciones ... 592
Permisos necesarios ... 593
Etiquetado de colecciones (consola) ... 593
Etiquetado de colecciones (AWS CLI) ... 594
Operaciones y complementos compatibles ... 594
Permisos y operaciones de la API de OpenSearch ... 595
Complementos de OpenSearch compatibles ... 604
Monitoreo de OpenSearch sin servidor ... 605
Monitoreo con CloudWatch ... 606
Monitoreo con CloudTrail ... 611
Monitoreo con EventBridge ... 617
Creación y administración de dominios ... 621
Creación de dominios de OpenSearch Service ... 621
Creación de dominios de OpenSearch Service (consola) ... 621
Creación de dominios de OpenSearch Service (AWS CLI) ... 628
Creación de dominios de OpenSearch Service (SDK de AWS) ... 630
Creación de dominios de OpenSearch Service (AWS CloudFormation) ... 630
Configuración de políticas de acceso ... 631
Migración a OpenSearch Service con el Asistente de Migración ... 631
Descripción general ... 632
Escenarios de migración ... 633
Introducción del Asistente de Migración ... 633
Arquitectura de soluciones ... 634
Configuración avanzada de clústeres ... 634
Cambios de configuración ... 635
Cambios que suelen causar implementaciones azul/verde ... 636
Cambios que no suelen causar implementaciones azul/verde ... 637
Cómo determinar si un cambio provocará una implementación azul/verde ... 638
Seguimiento de un cambio de configuración ... 642
Etapas de un cambio de configuración ... 644
Impacto en el rendimiento de las implementaciones azul/verde ... 648
Cargos por los cambios de configuración ... 648
Solución de errores de validación ... 649

Actualizaciones del software de servicio ... 655
Actualizaciones opcionales frente a actualizaciones obligatorias ... 655
Actualizaciones de revisión ... 656
Consideraciones ... 657
Iniciar una actualización ... 657
Intervalos de menor demanda ... 661
Supervisar actualizaciones ... 662
Cuando los dominios no son aptos para una actualización ... 662
Intervalos de menor demanda ... 663
Actualizaciones de software de servicio de menor demanda ... 664
Optimizaciones de ajuste automático durante horas de menor demanda ... 665
Habilitación del intervalo de menor demanda ... 666
Configuración de un intervalo de menor demanda personalizado ... 666
Visualización de acciones programadas ... 667
Reprogramación de acciones ... 669
Migración desde los intervalos de mantenimiento de ajuste automático ... 671
Notificaciones ... 672
Introducción a las notificaciones ... 672
Severidad de las notificaciones ... 673
Ejemplo de evento de EventBridge ... 674
Configuración de un dominio multi-AZ ... 675
Multi-AZ con modo de espera ... 675
Multi-AZ sin modo de espera ... 677
Interrupción de las zonas de disponibilidad ... 682
Compatibilidad con VPC ... 684
VPC frente a dominios públicos ... 684
Limitaciones ... 685
Arquitectura ... 685
Crear instantáneas de índice ... 693
Requisitos previos ... 694
Registrar un repositorio de instantáneas manuales ... 698
Tomar instantáneas manuales ... 702
Restaurar instantáneas ... 704
Eliminar instantáneas manuales ... 707
Automatizar instantáneas con la administración de instantáneas ... 707
Automatizar instantáneas con la administración de estado de índice ... 709

Utilizar Curator para instantáneas ... 709
Actualización de dominios ... 710
Rutas de actualización admitidas ... 710
Actualización de un dominio (consola) ... 715
Actualización de un dominio (CLI) ... 715
Actualización de un dominio (SDK) ... 716
Solución de errores de validación ... 717
Solución de problemas de una actualización ... 718
Uso de una instantánea para migrar datos ... 721
Creación de un punto de conexión personalizado ... 728
Puntos de conexión personalizados para nuevos dominios ... 729
Puntos de conexión personalizados para dominios existentes ... 730
Asignación CNAME ... 730
Ajuste automático ... 731
Tipos de cambios ... 731
Habilitar o deshabilitar la función de ajuste automático ... 733
Programación de mejoras de ajuste automático ... 734
Supervisión de cambios de ajuste automático ... 735
Etiquetado de dominios ... 735
Ejemplos de etiquetas ... 736
Etiquetado de dominios (consola) ... 737
Etiquetado de dominios (AWS CLI) ... 737
Etiquetado de dominios (AWS SDK) ... 739
Ejecución de acciones administrativas ... 740
Reiniciar el proceso de OpenSearch en un nodo de datos ... 741
Reinicio de un nodo de datos ... 741
Reiniciar el proceso de Dashboards ... 742
Limitaciones ... 743
Uso de consultas directas ... 744
Precios ... 744
Limitaciones ... 745
Limitaciones generales ... 745
Limitaciones de Amazon S3 ... 745
Limitaciones de Registros de Amazon CloudWatch ... 746
Limitaciones de Amazon Security Lake ... 746
Recomendaciones ... 747

Recomendaciones generales ... 747
Recomendaciones para Amazon S3 ... 748
Recomendaciones para Registros de CloudWatch ... 748
Recomendaciones para Security Lake ... 749
Cuotas ... 750
Cuotas para Amazon S3 ... 750
Cuotas para Registros de CloudWatch ... 751
Cuotas para Security Lake ... 752
Regiones de AWS compatible ... 753
Regiones de AWS disponibles para Amazon S3 ... 754
Regiones de AWS disponibles para Registros de CloudWatch ... 754
Regiones de AWS disponibles para Security Lake ... 755
Consultas directas en S3 ... 755
Creación de un origen de datos de S3 ... 756
Configuración de un origen de datos de S3 ... 764
Consultas directas en Registros de CloudWatch ... 767
Creación de un origen de datos de Registros de CloudWatch ... 767
Configuración de un origen de datos de Registros de CloudWatch ... 773
Consultas directas en Security Lake ... 775
Creación de un origen de datos de Security Lake ... 775
Configuración de un origen de datos de Security Lake ... 782
Administración de un origen de datos ... 785
Supervisión con orígenes de datos de métricas de CloudWatch ... 785
Activación y desactivación de orígenes de datos ... 789
Supervisión con AWS Budgets ... 789
Eliminación de un origen de datos ... 790
Optimización del rendimiento de las consultas ... 791
Índices de omisión ... 791
Vistas materializadas ... 792
Índices de cobertura ... 792
Comandos SQL y PPL compatibles ... 792
Comandos SQL compatibles ... 794
Comandos compatibles PPL ... 1002
Supervisión de dominios ... 1187
Monitoreo de métricas del clúster ... 1188
Ver métricas con CloudWatch ... 1189

Interpretación de gráficos de estado en OpenSearch Service 1189
Métricas de clúster 1190
Métricas de nodo maestro dedicado 1199
Métricas de nodos coordinadores dedicados 1200
Métricas de volumen de EBS 1201
Métricas de la instancia 1204
Métricas de UltraWarm 1217
Métricas de almacenamiento en frío 1223
Métricas de instancias optimizadas de OpenSearch (OR1) 1225
Métricas de alertas 1225
Métricas de detección de anomalías 1227
Métricas de búsqueda asíncrona 1229
Métricas de ajuste automático 1231
Métricas de Multi-AZ con modo de espera 1231
Métricas de un momento dado 1234
Métricas de SQL 1235
Métricas k-NN 1236
Métricas de búsqueda entre clústeres 1239
Métricas de replicación entre clústeres 1240
Aprender a clasificar métricas 1242
Métricas del lenguaje de procesamiento de canalizaciones 1242
Monitoreo de registros 1243
Habilitación de la publicación de registros (consola) 1245
Habilitación de la publicación de registros (AWS CLI) 1247
Habilitación de la publicación de registros (SDK de AWS) 1249
Habilitar publicación de registros (CloudFormation) 1250
Establecimiento de umbrales de registros lentos de solicitudes de búsqueda 1252
Establecimiento de umbrales de registros lentos de particiones 1252
Prueba de registros lentos 1252
Visualización de registros 1253
Monitoreo de registros de auditoría 1254
Limitaciones 1255
Habilitación de los registros de auditoría 1255
Habilite el registro de auditoría con la AWS CLI 1257
Habilite el registro de auditoría con la API de configuración 1257
Categorías y capas de registro de auditoría 1258

Configuración de registros de auditoría ... 1260
Ejemplo de registro de auditoría ... 1264
Configuración de registros de auditoría mediante la API REST ... 1267
Supervisión de eventos ... 1268
Eventos de actualización del software de servicio ... 1269
Eventos de ajuste automático ... 1276
Eventos del estado del clúster ... 1281
Eventos del punto de conexión de VPC ... 1294
Eventos de retirada de nodos ... 1297
Eventos de retirada de nodos degradados ... 1299
Eventos de error de dominio ... 1301
Tutorial: Listening for OpenSearch Service events ... 1303
Tutorial: Sending SNS alerts for available updates ... 1305
Monitoreo con CloudTrail ... 1307
Información de Amazon OpenSearch Service en CloudTrail ... 612
Descripción de las entradas del archivo de registro de Amazon OpenSearch Service ... 616
Seguridad ... 1312
Protección de los datos ... 1313
Cifrado en reposo ... 1314
Cifrado de nodo a nodo ... 1318
Identity and Access Management ... 1319
Tipos de políticas ... 1319
Realizar y firmar solicitudes de OpenSearch Service ... 1329
Cuando las políticas chocan ... 1330
Referencia de los elementos de las políticas ... 1331
Opciones avanzadas y consideraciones de la API ... 1337
Configurar políticas de acceso ... 1341
Políticas de muestra adicionales ... 1341
Referencia de permisos de la API ... 1341
AWSPolíticas administradas de ... 1342
Prevención de la sustitución confusa entre servicios ... 1353
Control de acceso detallado ... 1355
Panorámica general: control de acceso detallado y seguridad de OpenSearch Service ... 1356
Conceptos clave ... 1359
Acerca del usuario maestro ... 1360
Habilitar el control de acceso detallado ... 1361

Acceder a OpenSearch Dashboards como usuario maestro 1365
Administrar permisos 1367
Configuraciones recomendadas 1373
Limitaciones 1376
Modificar el usuario maestro 1378
Usuarios maestros adicionales 1378
Instantáneas manuales 1380
Integraciones 1380
Diferencias en la API de REST 1381
Tutorial: Control de acceso detallado con la autenticación de Cognito 1383
Tutorial: Base de datos de usuarios interna con autenticación básica 1388
Validación de conformidad 1391
Resiliencia 1392
JSON Web Tokens 1392
Consideraciones 1392
Modificación de la política de acceso al dominio 1393
Configuración de la autenticación y la autorización de JWT 1393
Uso de un JWT para enviar una solicitud de prueba 1394
Seguridad de la infraestructura 1396
Uso de puntos de conexión de VPC administrados por OpenSearch Service 1397
Autenticación SAML para OpenSearch Dashboards 1401
Información general de la configuración de SAML 1402
Consideraciones 1402
Autenticación SAML para dominios de VPC 1403
Modificación de la política de acceso al dominio 1403
Configuración de la autenticación iniciada por proveedor de servicios o por proveedor de identidades 1405
Configuración de la autenticación iniciada por proveedor de servicios y por proveedor de identidades 1412
Configuración de la autenticación SAML (AWS CLI) 1412
Configuración de la autenticación SAML (API de configuración) 1413
Solución de problemas de SAML 1413
Deshabilitar la autenticación SAML 1417
Compatibilidad del Centro de identidades de IAM con OpenSearch 1418
Consideraciones 1418
Modificación de la política de acceso al dominio 1419

Configuración de la autenticación y autorización de Centro de identidades de IAM (consola) ... 1419
Configuración del control de acceso detallado ... 1420
Configuración de la autenticación y autorización del Centro de identidades de IAM (CLI) ... 1420
Deshabilitación de la autenticación del Centro de identidades de IAM en el dominio ... 1421
Descripción de la autenticación de Amazon Cognito para OpenSearch Dashboards ... 1421
Requisitos previos ... 1422
Configuración de un dominio para utilizar la autenticación de Amazon Cognito ... 1426
Permitir el rol autenticado ... 1430
Configuración de proveedores de identidades ... 1431
(Opcional) Configuración de acceso pormenorizado ... 1431
(Opcional) Personalización de la página de inicio de sesión ... 1433
(Opcional) Configuración de seguridad avanzada ... 1433
Testeo ... 1433
Cuotas ... 1433
Problemas habituales de configuración ... 1434
Deshabilitar la autenticación de Amazon Cognito para OpenSearch Dashboards ... 1439
Eliminación de dominios que utilizan la autenticación de Amazon Cognito para OpenSearch Dashboards ... 1439
Cómo utilizar roles vinculados a servicios ... 1439
Rol de creación de dominios de VPC y orígenes de datos ... 1440
Rol de creación de colecciones ... 1443
Rol de creación de canalizaciones ... 1446
Código de muestra ... 1450
Compatibilidad con clientes de Elasticsearch ... 1450
Compresión de solicitudes HTTP ... 1451
Habilitar la compresión gzip ... 1451
Encabezados obligatorios ... 1452
Código de muestra (Python 3) ... 1452
Utilizar los SDK de AWS ... 1454
Java ... 1454
Python ... 1465
Nodo ... 1468
Indexación de datos ... 1471
Restricciones de nomenclatura de los índices ... 1471
Reducción del tamaño de la respuesta ... 1472

Códecs de índice ... 1474
Carga de datos de streaming en OpenSearch Service ... 1474
Carga de datos de streaming desde OpenSearch Ingestion ... 1475
Carga de datos de streaming desde Amazon S3 ... 1475
Cargar datos de streaming desde Amazon Kinesis Data Streams ... 1481
Carga de datos de streaming desde Amazon DynamoDB ... 1485
Carga de datos de streaming desde Amazon Data Firehose ... 1490
Cargar datos de streaming desde Amazon CloudWatch ... 1491
Carga de datos de streaming desde AWS IoT ... 1491
Carga de datos con Logstash ... 1491
Configuración ... 1491
Buscar datos ... 1495
Búsquedas de URI ... 1495
Búsquedas de cuerpo de la solicitud ... 1497
Potenciar campos ... 1499
Resaltado de resultados de búsqueda ... 1499
API de recuento ... 1501
Paginar los resultados de búsqueda ... 1502
Punto en el tiempo ... 1502
Los parámetros `from` y `size` ... 1502
Lenguaje de consulta de paneles ... 1503
Paquetes ... 1505
Permisos necesarios ... 1506
Carga de paquetes en Amazon S3 ... 1506
Importación y asociación de paquetes ... 1507
Uso de paquetes con OpenSearch ... 1508
Actualización de paquetes ... 1513
Actualización manual de los índices con un nuevo diccionario ... 1516
Disociación y eliminación de paquetes ... 1518
Complementos personalizados ... 1519
Complementos de terceros ... 1536
Compatibilidad con SQL ... 1540
Ejemplo de llamada ... 1542
Notas y diferencias ... 1543
SQL Workbench ... 1544
CLI SQL ... 1395

Controlador JDBC 1544
Controlador ODBC 1544
Búsqueda en clústeres 1544
Limitaciones 1545
Requisitos previos de búsqueda entre clústeres 1546
Precio de búsqueda entre clústeres 1546
Configuración de una conexión 1546
Eliminación de una conexión 1548
Configuración de seguridad y explicación de ejemplo 1548
OpenSearch Dashboards 1555
Aprender a clasificar 1555
Introducción a Aprender a clasificar 1556
API de Aprender a clasificar 1578
Búsqueda asíncrona 1584
Ejemplo de llamada de búsqueda 1584
Permisos de búsqueda asíncrona 1586
Configuración de búsqueda asincrónica 1587
Búsqueda en clústeres 1587
UltraWarm 1589
Punto en el tiempo 1589
Consideraciones 1590
Creación de un PIT 1590
Permisos de puntos en el tiempo 1592
Configuración de PIT 1593
Búsqueda en clústeres 1593
UltraWarm 1594
Búsqueda semántica 1594
Búsqueda de segmentos simultánea 1594
Generación de consultas en lenguaje natural 1595
Requisitos previos 1595
Introducción 1596
Configuración de permisos 1596
Automatización de la configuración 1597
Búsqueda vectorial 1598
(Versión preliminar) Integración de OpenSearch Service con Amazon S3 Vectors 1599
(Versión preliminar) Importación desde Amazon S3 Vectors a OpenSearch sin servidor 1599

(Vista previa) Capacidades de búsqueda avanzada con un motor de Amazon S3 Vectors . 1608
Búsqueda k-NN 1615
Introducción a k-NN 1616
Diferencias, ajuste y limitaciones de k-NN 1618
OpenSearch Dashboards 1620
Control del acceso a Dashboards 1621
Utilización de un proxy para acceder a OpenSearch Service desde Dashboards 1621
Configuración de OpenSearch Dashboards para utilizar un servidor de mapas de WMS 1625
Conexión de un servidor local de Dashboards a OpenSearch Service 1626
Administración de índices en OpenSearch Dashboards 1628
Características adicionales 1628
OpenSearch UI 1629
Historial de versiones 1630
Introducción 1632
Permisos necesarios para crear aplicaciones de Amazon OpenSearch Service 1632
Creación de una aplicación 1636
Administrar los administradores de aplicaciones 1644
Habilitar la federación SAML con IAM 1647
Paso 1: Configurar la aplicación de proveedor de identidades (Okta) 1648
Paso 2: Configurar AWS para Okta 1652
Paso 3: Creación de la política de acceso de Amazon OpenSearch Service en IAM 1654
Paso 4: Compruebe la experiencia de inicio de sesión único iniciada por el proveedor de identidad con SAML 1657
Paso 5: Configuración del control de acceso detallado basado en atributos de SAML 1661
Administración de las asociaciones de orígenes de datos y los permisos de acceso a la VPC 1665
Asociación de un origen de datos a una aplicación de OpenSearch UI 1665
Administrar el acceso a los dominios en una VPC 1666
Configuración del acceso a las colecciones de OpenSearch sin servidor en una VPC 1667
Uso de los espacios de trabajo de Amazon OpenSearch Service 1671
Creación de espacios de trabajo para aplicaciones de OpenSearch UI 1671
La privacidad del espacio de trabajo y los colaboradores 1672
Tipos de WorkSpace 1673
Acceso a datos entre regiones y entre cuentas con búsqueda entre clústeres 1674
Configuración de permisos de acceso para el acceso a datos entre regiones y cuentas mediante la búsqueda entre clústeres 1676
Crear una conexión entre dominios 1679

Probar su configuración de seguridad para el acceso a datos entre cuentas y regiones con búsqueda entre clústeres ... 1681
Eliminación de una conexión ... 1683
Administración del acceso a OpenSearch UI desde un punto de conexión de VPC ... 1684
Creación de una conexión privada entre una VPC y OpenSearch UI ... 1684
Actualización de la política de puntos de conexión de VPC para permitir el acceso a la aplicación de OpenSearch UI ... 1686
Revocación del acceso a OpenSearch UI en una política de punto de conexión de VPC ... 1687
Cuotas y puntos de conexión ... 1687
Puntos de conexión de OpenSearch UI ... 1688
Service Quotas de OpenSearch UI ... 1691
Administración de índices ... 1692
Almacenamiento UltraWarm ... 1692
Requisitos previos ... 1693
Requisitos de almacenamiento UltraWarm y consideraciones de rendimiento ... 1695
Precios de UltraWarm ... 1696
Habilitar UltraWarm ... 1696
Migración de índices al almacenamiento UltraWarm ... 1699
Automatizar migraciones ... 1702
Ajuste de migración ... 1702
Cancelación de migraciones ... 1703
Listado de índices calientes y templados ... 1703
Devolución de índices templados al almacenamiento caliente ... 1703
Restauración de índices templados a partir de instantáneas ... 1704
Instantáneas manuales de índices templados ... 1705
Migración de índices templados al almacenamiento frío ... 1706
Prácticas recomendadas para índices KNN ... 1707
Deshabilitar UltraWarm ... 1707
Almacenamiento en frío ... 1708
Requisitos previos ... 1709
Requisitos de almacenamiento en frío y consideraciones de rendimiento ... 1710
Precio del almacenamiento en frío ... 1711
Habilitación del almacenamiento en frío ... 1711
Administración de índices fríos en OpenSearch Dashboards ... 1713
Migración de índices al almacenamiento frío ... 1713
Automatización de las migraciones al almacenamiento en frío ... 1715

Cancelación de migraciones al almacenamiento en frío ... 1715
Listado de índices almacenados en frío ... 1716
Migración de índices fríos al almacenamiento templado ... 1720
Restauración de índices fríos a partir de instantáneas ... 1721
Cancelación de migraciones del almacenamiento en frío al templado ... 1721
Actualización de metadatos de índices almacenados en frío ... 1722
Eliminación de índices almacenados en frío ... 1722
Deshabilitar el almacenamiento en frío ... 1723
Instancias optimizadas para OpenSearch ... 1723
Limitaciones ... 1724
Ajustes para mejorar el rendimiento de la ingesta ... 1724
En qué se diferencian las instancias optimizadas para OpenSearch de otras instancias ... 1725
En qué se diferencian las instancias optimizadas para OpenSearch de las instancias UltraWarm ... 1725
Aprovisionamiento de un dominio con instancias optimizadas de OpenSearch ... 1726
Administración de estados de índice ... 1727
Crear una política de ISM ... 1728
Ejemplos de política ... 1729
Plantillas de ISM ... 1733
Diferencias ... 1734
Tutorial: Automatización de procesos de ISM ... 1735
Acumulaciones de índices ... 1740
Crear un trabajo acumulativo de índices ... 1740
Transformaciones de índices ... 1742
Creación de un trabajo de transformación de índice ... 1742
Replicación entre clústeres ... 1744
Limitaciones ... 1745
Requisitos previos ... 1745
Requisitos de los permisos ... 1746
Configuración de una conexión entre clústeres ... 1747
Inicio de la replicación ... 1748
Confirmación de replicación ... 1749
Pausa y reanudación de la replicación ... 1750
Detención de la replicación ... 1751
Seguimiento automático ... 1751
Actualización de los dominios conectados ... 1753

Reindexación remota 1753
Requisitos previos 1754
Reindexación de datos entre dominios de internet de OpenSearch Service 1754
Reindexación de datos cuando el dominio remoto esté en una VPC 1756
Reindexar datos entre dominios que no son de OpenSearch Service 1760
Reindexar conjuntos de datos grandes 1761
Configuración remota de reindexación 1763
Data Streams 1763
Introducción a Data Streams 1764
Monitoreo de datos 1768
Alertas 1768
Permisos de alertas 1769
Introducción a las alertas 1769
Notificaciones 1770
Diferencias 1771
Detección de anomalías 1772
..... 1773
Tutorial: Detectar un uso elevado de la CPU con la detección de anomalías 1776
Compatibilidad con Amazon Q 1780
Regiones de AWS compatible 1781
Configuración de Amazon Q para OpenSearch Service 1781
Genere visualizaciones utilizando un lenguaje natural 1782
Ver resúmenes e información sobre alertas 1783
Antes de empezar 1786
Visualización de resúmenes de alertas y conocimientos 1787
Consulta los resúmenes de los resultados de las consultas generados por Amazon Q en la página Discover 1788
Vea los detectores de anomalías recomendados 1790
Acceda al chat de Amazon Q para preguntas sobre OpenSearch Service 1791
Machine learning 1795
Conectores para Servicios de AWS 1795
Requisitos previos 1796
Cómo crear un conector de OpenSearch Service 1799
Conectores para plataformas externas 1801
Requisitos previos 1802
Cómo crear un conector de OpenSearch Service 1805

Integraciones de plantilla de CloudFormation ... 1807
Plantillas CloudFormation disponibles ... 1808
Requisitos previos ... 1809
Plantillas de Amazon Bedrock ... 1811
Plantillas de integración de servidores MCP ... 1812
Plantillas Amazon SageMaker ... 1815
Inferencia remota para plantillas de resaltado semántico ... 1816
Configuración de ML Commons no compatible ... 1817
Complemento de Flow Framework ... 1818
Creación de conectores de ML en OpenSearch Service ... 1818
Configuración de permisos ... 1826
Security Analytics ... 1828
Componentes y conceptos de Security Analytics ... 1828
Tipos de registro ... 1829
Detectores ... 1829
Reglas ... 1829
Resultados ... 1830
Alertas ... 1830
Exploración de Security Analytics ... 1830
Configuración de permisos ... 1832
Solución de problemas ... 1834
No existe tal error de índice ... 1834
Observabilidad ... 1835
Explore los datos con análisis de eventos ... 1835
Crear visualizaciones ... 1838
Profundizar más con Trace Analytics ... 1839
Análisis de seguimiento ... 1839
Requisitos previos ... 1840
Configuración de ejemplo de OpenTelemetry Collector ... 1841
Configuración de muestra de OpenSearch Ingestion ... 1842
Exploración de datos de rastreo ... 1843
Lenguaje de procesamiento de canalizaciones ... 1844
... 1845
Prácticas recomendadas ... 1847
Monitorización y alertas ... 1847
Configuración de alarmas de CloudWatch ... 1847

Habilitación de la publicación de registros ... 1848
Estrategia de particiones ... 1848
Determinar los recuentos de particiones y nodos de datos ... 1849
Evite el sesgo de almacenamiento ... 1850
Stability ... 1850
Manténgase al día con OpenSearch ... 1850
Cómo mejorar el rendimiento de las instantáneas ... 1851
Habilite los nodos maestros dedicados ... 1852
Implemente en varias zonas de disponibilidad ... 1852
Controle el flujo de incorporación y el almacenamiento en búfer ... 1852
Cree asignaciones para cargas de trabajo de búsqueda ... 1853
Utilice plantillas de índice ... 1854
Administre los índices con la administración de estado de índice ... 1855
Eliminar los índices que no se utilizan ... 1855
Utilice varios dominios para disfrutar de una alta disponibilidad ... 1855
Rendimiento ... 1856
Optimice el tamaño y la compresión de solicitudes ... 1856
Reduzca el tamaño de las respuestas de solicitudes masivas ... 1856
Ajuste intervalos de actualización ... 1857
Habilite el ajuste automático ... 1857
Seguridad ... 1857
Cómo habilitar el control de acceso detallado ... 1857
Implemente dominios dentro de una VPC ... 1858
Aplique una política de acceso restrictivo ... 1858
Habilite el cifrado en reposo ... 1858
Habilite el cifrado de nodo a nodo ... 1859
Monitorización de con AWS Security Hub CSPM ... 1859
Optimización de costos ... 1859
Use los tipos de instancia de última generación ... 1859
Use los volúmenes gp3 de Amazon EBS más recientes ... 1860
Utilice almacenamiento en frío y UltraWarm para datos de registro de series temporales ... 1860
Revise las recomendaciones para instancias reservadas ... 1861
Determinación del tamaño de dominios ... 1861
Cálculo de requisitos de almacenamiento ... 1861
Selección del número de particiones ... 1863
Selección de tipos de instancias y pruebas ... 1865

Escala de petabytes 1867
Nodos coordinadores dedicados 1869
Cuándo utilizar nodos coordinadores dedicados 1869
Arquitectura y comportamiento 1869
Requisitos y limitaciones 1870
Aprovisionamiento de nodos coordinadores dedicados 1870
Prácticas recomendadas 1871
Nodos maestros dedicados 1873
Elección del número de nodos maestros dedicados 1874
Elección de tipos de instancias para nodos principales dedicados 1876
Alarmas de CloudWatch recomendadas 1877
Referencia general 1885
Tipos de instancias admitidas 1885
Tipos de instancias de generación actual 1885
Tipos de instancias de generación anterior 1910
Características por versión de motor 1914
Complementos por versión de motor 1920
Complementos opcionales 1925
Operaciones admitidas 1925
Diferencias de API destacadas 1926
Cuotas 1982
Cuotas de almacenamiento UltraWarm 1982
Número de nodos de datos por AZ 1876
Límite total de nodos por familia de instancias 1983
Cuotas de tamaño del volumen de EBS 1984
Cuotas de red 1996
Cuotas de tamaño de la partición 2005
Cuotas de recuento de la partición 2006
Cuota de procesamiento de Java 2006
Cuota de políticas de dominio 2006
instancias reservadas 2007
Comprar instancias reservadas (consola) 2007
Comprar instancias reservadas (CLI de AWS) 2008
Comprar instancias reservadas (SDK de AWS) 2011
Examinar los costos 2013
Otros recursos admitidos 2013

Tutoriales 2015
Creación y búsqueda de documentos 2015
Requisitos previos 2015
Adición de un documento a un índice 2016
Creación de ID de generación automática 2017
Actualización de un documento con un comando POST 2018
Ejecución de acciones por lotes 2019
Búsqueda de documentos 2020
Recursos relacionados 2022
Migrar a OpenSearch Service 2022
Tomar y cargar la instantánea 2022
Crear un dominio 2024
Conceda permisos al bucket de S3. 2025
Restaurar la instantánea 2027
Creación de una aplicación de búsqueda 2030
Requisitos previos 2031
Paso 1: Indexe los datos de muestra 2031
Paso 2: crear e implementar una función de Lambda 2032
Paso 3: crear la API en API Gateway 2035
Paso 4: (opcional) modificar la política de acceso a dominios 2037
Asigne el rol de Lambda (si utiliza un control de acceso detallado) 2040
Paso 5: Pruebe la aplicación web 2041
Pasos a seguir a continuación 2043
Visualización de las llamadas del servicio de atención 2044
Paso 1: configure los requisitos previos 2045
Paso 2: copie el código de muestra 2046
(Opcional) Paso 3: indexe los datos de ejemplo 2050
Paso 4: analice y visualice sus datos 2052
Paso 5: elimine recursos y pasos siguientes 2056
Cambio de nombre de Amazon OpenSearch Service 2058
Nueva versión de la API 2058
Tipos de instancia con cambio de nombre 2059
Cambios en las políticas de acceso 2059
Políticas de IAM 2059
Políticas de SCP 2059
Nuevos tipos de recursos 2060

Kibana cambió de nombre a OpenSearch Dashboards 2061
Métricas de CloudWatch con cambio de nombre 2062
Cambios en la consola de Billing and Cost Management 2063
Nuevo formato de evento 2064
¿Qué permanece igual? 2064
Comience: Actualice sus dominios a OpenSearch 1.x 2064
Solución de problemas 2066
No se puede acceder a OpenSearch Dashboards 2066
No se puede obtener acceso al dominio de VPC 2066
Clúster en estado de solo lectura 2066
Estado rojo del clúster 2068
Corrección automática de clústeres en rojo 2069
Recuperación de una carga de procesamiento elevada continua 2070
Estado amarillo del clúster 2072
ClusterBlockException 2072
Falta de espacio de almacenamiento disponible 2073
Presión alta de memoria de JVM 2073
Error al migrar a multi-AZ con modo de espera 2074
Creación de un índice, una plantilla de índice o una política de ISM durante la migración de dominios sin modo de espera a dominios con modo de espera 1834
Número incorrecto de copias de datos 2074
OutOfMemoryError de JVM 2075
Nodos de clúster defectuosos 2076
Límite máximo de fragmentos superado 2076
Dominio atascado en estado de procesamiento 2077
Bajo balance de ráfaga EBS 2077
La métrica de EBS aumenta durante el redimensionamiento del volumen 2078
No se pueden habilitar los registros de auditoría 2078
No se puede cerrar el índice 2079
Verificaciones de licencias del cliente 2079
Limitación controlada de solicitudes 2079
No se puede usar SSH en nodo 2079
Error de instantánea "No válido para la clase de almacenamiento del objeto" 2079
Encabezado de host no válido 2080
Tipo de instancia M3 no válido 2080
Las consultas activas dejan de funcionar después de habilitar UltraWarm 2080

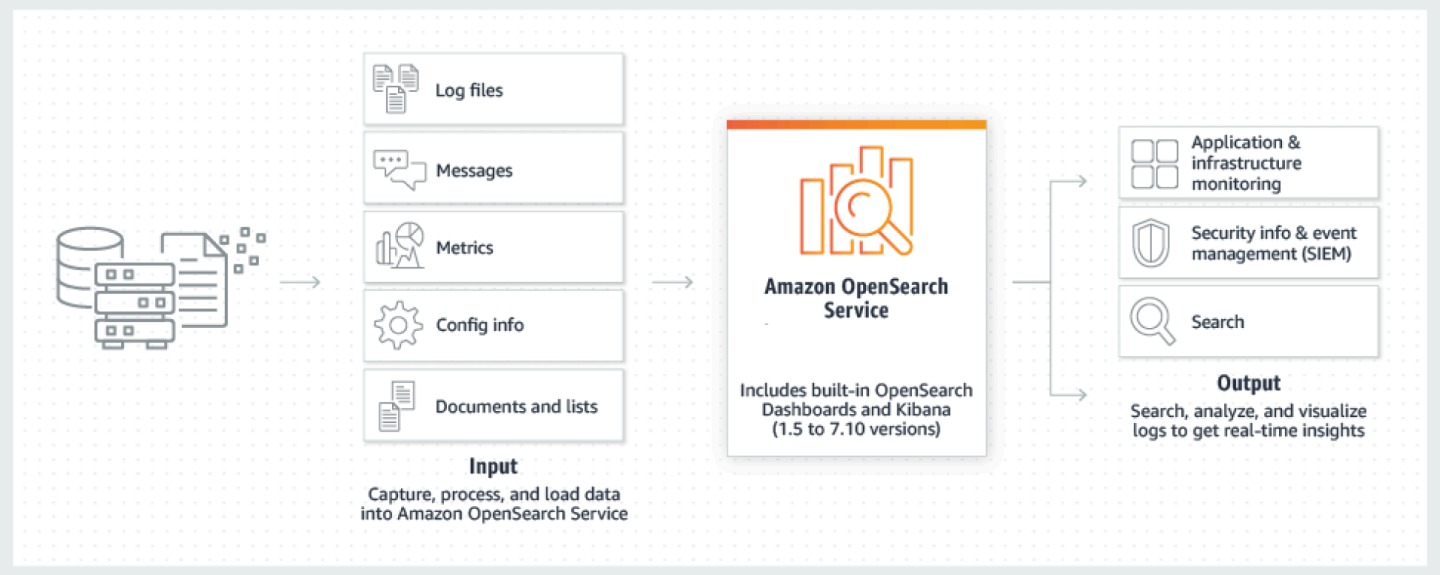
No se puede cambiar a una versión anterior después de una actualización 2081
Resumen necesario de dominios para todas las Regiones de AWS 2081
Error del navegador al usar paneles de OpenSearch 2082
Partición de nodos y sesgo de almacenamiento 2082
Partición del índice y el sesgo de almacenamiento 2083
Operación no autorizada tras seleccionar acceso a la VPC 2084
Bloqueo al cargar después de crear el dominio de la VPC 2084
Solicitudes denegadas a la API de OpenSearch 2084
No se puede conectar desde Alpine Linux 2085
Hay demasiadas solicitudes de Search Backpressure 2086
Error de certificado cuando se utiliza un SDK 2086
La instalación del complemento personalizado falla debido a la compatibilidad de las versiones 2087
Historial de revisión 2090
Actualizaciones anteriores 2160
AWSGlosario de 2164

¿Qué es Amazon OpenSearch Service?

Amazon OpenSearch Service es un servicio administrado que facilita la implementación, la operación y el escalado de clústeres de OpenSearch en la nube de AWS. Un dominio de OpenSearch Service es sinónimo de clúster de OpenSearch. Los dominios son clústeres con la configuración, los tipos de instancia, los recuentos de instancias y los recursos de almacenamiento que especifique. Amazon OpenSearch Service es compatible con OpenSearch y con la version heredada de Elasticsearch OSS (con hasta 7.10, la versión final de código abierto del software). Al crear un dominio, tiene la opción de elegir qué motor de búsqueda utilizar.

OpenSearch es un motor de búsqueda y análisis totalmente de código abierto para casos de uso como análisis de registros, monitoreo de aplicaciones en tiempo real y análisis de secuencias de clics. Para más información, consulte Documentación de OpenSearch.

Amazon OpenSearch Service le proporciona todos los recursos para su clúster y lo inicia. También detecta y sustituye automáticamente los nodos de OpenSearch Service que tienen algún error. De este modo, reduce la sobrecarga asociada con las infraestructuras autoadministradas. Puede escalar el clúster con una única llamada al API o con algunos clics en la consola.



Para empezar a usar OpenSearch Service, debe crear un dominio de OpenSearch Service, que equivale a un clúster de OpenSearch Service. Cada instancia de EC2 del clúster actúa como un nodo de OpenSearch Service.

Puede utilizar la consola de OpenSearch Service para instalar y configurar un dominio en unos minutos. Si prefiere el acceso mediante programación, puede utilizar la AWS CLI, los AWS SDK o Terraform.

Características de Amazon OpenSearch Service

OpenSearch Service incluye las siguientes características:

Escalado

- Varias configuraciones de CPU, memoria y capacidad de almacenamiento, que se denominan tipos de instancias, que incluyen instancias rentables de Graviton
- Soporta hasta 1002 nodos de datos
- Hasta 25 PB de almacenamiento asociado
- Almacenamiento rentable UltraWarm y frío para datos de solo lectura

Seguridad

- Control de acceso de AWS Identity and Access Management (IAM)
- Integración sencilla con Amazon VPC y grupos de seguridad de VPC
- Cifrado de datos en reposo y cifrado de nodo a nodo.
- Autenticación de Amazon Cognito, HTTP Basic o SAML para paneles de OpenSearch
- Seguridad en el nivel de índice, de documento y de campo
- Registros de auditoría
- Tenencia múltiple de paneles

Stability

- Numerosas ubicaciones geográficas para los recursos, denominadas regiones y zonas de disponibilidad
- Asignación de nodos en dos o tres zonas de disponibilidad de la misma región de AWS, lo que se conoce como Multi-AZ
- Nodos principales dedicados para aligerar las tareas de administración del clúster
- Instantáneas automatizadas para realizar backups y restaurar dominios de OpenSearch Service

Flexibilidad

- Compatibilidad con SQL para la integración con aplicaciones de inteligencia empresarial (BI)
- Paquetes personalizados para mejorar los resultados de búsqueda

Integración con servicios populares

- Visualización de datos mediante OpenSearch Dashboards
- Integración con Amazon CloudWatch para monitorear las métricas de los dominios de OpenSearch Service y definir alarmas
- Integración con AWS CloudTrail para auditar las llamadas a la API de configuración en los dominios de OpenSearch Service
- Integración con Amazon S3, Amazon Kinesis y Amazon DynamoDB para cargar datos de streaming en OpenSearch Service
- Alertas de Amazon SNS cuando los datos superan determinados umbrales

Cuándo se debe usar OpenSearch en lugar de Amazon OpenSearch Service

Utilice la siguiente tabla para decidir si Amazon OpenSearch Service aprovisionado u OpenSearch autoadministrado es la opción más adecuada.

OpenSearch	Amazon OpenSearch Service
• Su organización está dispuesta a supervisar y mantener manualmente los clústeres autoaprovisionados, y cuenta con personas con las habilidad es adecuadas para ello. • Desea tener un control total del código a nivel de compilación. • Su organización prefiere, o utiliza de forma exclusiva, el software de código abierto.	• No desea administrar, supervisar y mantener su infraestructura manualmente. • Desea formas sencillas de administrar los crecientes costos de análisis mediante la organizac ión de capas de datos en distintos niveles de almacenamiento, lo que permite aprovechar la durabilidad y el bajo costo de Amazon S3. • Desea aprovechar las integraciones con otros Servicios de AWS como DynamoDB, Amazon DocumentDB (compatible con MongoDB), IAM, CloudWatch y CloudFormation.

OpenSearch	Amazon OpenSearch Service
• Tiene una estrategia multinube que requiere tecnologías que no son específicas de cada proveedor. • Su equipo es capaz de abordar cualquier problema crítico de producció n. • Desea disponer de la flexibilidad necesaria para utilizar, modificar y ampliar el producto de la forma que desee. • Desea tener acceso inmediato a las nuevas características tan pronto como se publiquen.	• Desea acceder fácilmente a la asistencia de Soporte si tiene problemas de mantenimiento preventivo o de producción. • Desea aprovechar características como la autorreparación, el mantenimiento proactivo, la resiliencia y las copias de seguridad.

Versiones compatibles de Elasticsearch y OpenSearch

OpenSearch Service admite las siguientes versiones de OpenSearch:

- 3.1, 2.19, 2.17, 2.15, 2.13, 2.11, 2.9, 2.7, 2.5, 2.3, 1.3, 1.2, 1.1 y 1.0

OpenSearch Service admite las siguientes versiones de Elasticsearch heredado:

- 7.10, 7.9, 7.8, 7.7, 7.4, 7.1, 6.8, 6.7, 6.5, 6.4, 6.3, 6.2, 6.0, 5.6, 5.5, 5.3, 5.1, 2.3 y 1.5

Recomendamos actualizar a la última versión disponible de OpenSearch para aprovechar al máximo el servicio OpenSearch, en términos de precio-rendimiento, riqueza de características y mejoras de seguridad.

Soporte extendido y estándar

AWS proporciona correcciones de errores y actualizaciones de seguridad para las versiones con soporte estándar. Para las versiones con soporte extendido, AWS ofrece correcciones de seguridad críticas durante al menos 12 meses después de que finalice el soporte estándar, a una tarifa fija por

hora de instancia normalizada (NIH). El NIH se basa en el tamaño de la instancia y en las horas de uso.

Los cargos de soporte extendido se aplican automáticamente cuando un dominio ejecuta una versión que ya no cuenta con el soporte estándar. Para evitar estos cargos, actualice a una versión compatible.

Las siguientes tablas muestran el calendario de finalización del soporte para OpenSearch y las versiones antiguas de Elasticsearch.

El Servicio OpenSearch admite muchas versiones de OpenSearch y versiones heredadas de Elasticsearch de código abierto. Para algunas versiones, ya publicamos las fechas de fin del soporte estándar y ampliamos el soporte. Recomendamos actualizar a la última versión disponible de OpenSearch para aprovechar al máximo el servicio OpenSearch, en términos de precio-rendimiento, riqueza de características y mejoras de seguridad. Las siguientes tablas proporcionan listas de las versiones de Elasticsearch y OpenSearch y sus programas de soporte.

El calendario de finalización del soporte para las versiones de Elasticsearch es el siguiente:

Versión del software	Fecha de finalización del soporte estándar	Fecha de finalización del soporte extendido
Elasticsearch versiones 1.5 y 2.3	7 de noviembre de 2025	7 de noviembre de 2026
Elasticsearch versiones de la 5.1 a la 5.5	7 de noviembre de 2025	7 de noviembre de 2026
Elasticsearch versión 5.6	7 de noviembre de 2025	7 de noviembre de 2028
Elasticsearch	7 de noviembre de 2025	7 de noviembre de 2026

Versión del software	Fecha de finalización del soporte estándar	Fecha de finalización del soporte extendido
versiones de la 6.0 a la 6.7		
Elasticsearch versión 6.8	No se ha anunciado	No se ha anunciado
Elasticsearch versiones de la 7.1 a la 7.8	7 de noviembre de 2025	7 de noviembre de 2026
Elasticsearch versión 7.9	No se ha anunciado	No se ha anunciado
Elasticsearch versión 7.10	No se ha anunciado	No se ha anunciado

El calendario de finalización del soporte para las versiones de OpenSearch es el siguiente:

Versión del software	Fecha de finalización del soporte estándar	Fecha de finalización del soporte extendido
Versiones 1.0 a 1.2 de OpenSearch	7 de noviembre de 2025	7 de noviembre de 2026

Versión del software	Fecha de finalización del soporte estándar	Fecha de finalización del soporte extendido
OpenSearch versión 1.3	No se ha anunciado	No se ha anunciado
OpenSearch versiones 2.3 a 2.9	7 de noviembre de 2025	7 de noviembre de 2026
OpenSearch versión 2.11 y posteriores	No se ha anunciado	No se ha anunciado

Soporte estándar y soporte extendido para OpenSearch y Elasticsearch

AWS proporciona correcciones de errores y actualizaciones de seguridad periódicas para las versiones incluidas en Standard Support. Para las versiones con soporte extendido, AWS proporciona correcciones de seguridad críticas durante un periodo de al menos 12 meses después de la finalización del soporte estándar, por una tarifa fija adicional por cada hora de instancia normalizada (NIH). El NIH se calcula como un factor del tamaño de la instancia (por ejemplo, mediana o grande) y del número de horas de la instancia (consulte la sección sobre el cálculo de los cargos por soporte prolongado que aparece a continuación para ver un ejemplo). Los cargos de soporte extendido se aplican automáticamente cuando un dominio ejecuta una versión para la que el soporte estándar ha finalizado. Puede actualizar a una versión reciente que aún esté cubierta por el soporte estándar para evitar los cargos por soporte extendido. Para obtener más información sobre los cargos por soporte extendido, consulte la página de precios. Para obtener información general sobre el soporte extendido, consulte las Preguntas frecuentes sobre soporte extendido.

Cálculo de los cargos por soporte extendido

A los dominios que ejecuten versiones con soporte ampliado se les cobrará una tarifa fija adicional por hora de instancia normalizada (NIH), por ejemplo, 0,0065 USD en la región EE. UU. Este (Virginia del Norte). El NIH se calcula como un factor del tamaño de la instancia (p. ej., mediana o grande) y del número de horas de la instancia. Por ejemplo, si ejecuta una instancia de m7g.medium.search durante 24 horas en la región EE. UU. Este (Virginia del Norte), cuyo precio es de 0,068 dólares/hora de instancia (bajo demanda), normalmente tendrá que pagar 1632 USD (0,068 USD x 24). Si ejecuta una versión con soporte extendido, pagará 0,0065 USD/NIH adicionales, que se calculan como 0,0065 USD x 24 (número de horas de instancia) x 2 (factor de normalización de tamaño; 2 para instancias medianas), lo que equivale a 0,312 USD por soporte extendido durante 24 horas. El importe total que pagará durante 24 horas será la suma del costo de uso de la instancia estándar y el costo del soporte extendido, que es de 1944 USD (1,632 USD + 0,312 USD). La siguiente tabla muestra el factor de normalización para varios tamaños de instancias en OpenSearch Service.

Tamaño de instancia	Factor de normalización
nano	0,25
micro	0,5
small	1
medium	2
large	4
xlarge	8
2xlarge	16
4xlarge	32
8xlarge	64
9xlarge	72

Tamaño de instancia	Factor de normalización	
10xlarge	80	
12xlarge	96	
16xlarge	128	
18xlarge	144	
24xlarge	192	
32xlarge	256	

Precios de Amazon OpenSearch Service

Para OpenSearch Service, paga por cada hora de uso de una instancia de EC2 y por el tamaño acumulado de cualquier volumen de almacenamiento de EBS adjunto a sus instancias. También se aplican cargos por transferencia de datos de AWS estándar.

Sin embargo, existen algunas excepciones notables en la transferencia de datos. Si un dominio utiliza varias zonas de disponibilidad, OpenSearch Service no facturará el tráfico entre estas zonas de disponibilidad. Se produce una transferencia de datos significativa dentro de un dominio durante la asignación de particiones y el reequilibrio. OpenSearch Service no mide ni factura este tráfico. Del mismo modo, OpenSearch Service no factura la transferencia de datos entre nodos UltraWarm/fríos y Amazon S3.

Para obtener información completa sobre precios, consulte los precios de Amazon OpenSearch Service. Para obtener información acerca de los gastos generados por los cambios de configuración, consulte the section called “Cargos por los cambios de configuración”.

Servicios relacionados

OpenSearch Service se suele utilizar con los siguientes servicios:

Amazon CloudWatch

Los dominios de OpenSearch Service envían métricas automáticamente a CloudWatch para que pueda monitorear el estado y el rendimiento del dominio. Para más información, consulte Monitoreo de métricas del clúster de OpenSearch con Amazon CloudWatch.

CloudWatch Logs también puede funcionar en la dirección opuesta. Es posible configurar CloudWatch Logs para transmitir datos a OpenSearch Service para su análisis. Para más información, consulte the section called “Cargar datos de streaming desde Amazon CloudWatch”.

AWS CloudTrail

Puede utilizar AWS CloudTrail para obtener un historial de las llamadas a la API de configuración de OpenSearch Service y los eventos relacionados de su cuenta. Para más información, consulte Monitoreo de las llamadas a la API de Amazon OpenSearch Service con AWS CloudTrail.

Amazon Kinesis

Kinesis es un servicio administrado para el procesamiento de datos de streaming en tiempo real a una escala masiva. Para más información, consulte the section called “Cargar datos de streaming desde Amazon Kinesis Data Streams” y the section called “Carga de datos de streaming desde Amazon Data Firehose”.

Amazon S3

Amazon Simple Storage Service (Amazon S3) proporciona almacenamiento para Internet. Esta guía proporciona el código de muestra de Lambda para la integración con Amazon S3. Para más información, consulte the section called “Carga de datos de streaming desde Amazon S3”.

AWS IAM de

AWS Identity and Access Management (IAM) es un servicio web que puede utilizar para administrar el acceso a sus dominios de OpenSearch Service. Para más información, consulte the section called “Identity and Access Management”.

AWS Lambda

AWS Lambda es un servicio automático que permite ejecutar código sin aprovisionar ni administrar servidores. Esta guía proporciona código de muestra de Lambda para transmitir datos desde DynamoDB, Amazon S3 y Kinesis. Para más información, consulte the section called “Carga de datos de streaming en OpenSearch Service”.

Amazon DynamoDB

Amazon DynamoDB es un servicio de base de datos NoSQL totalmente administrado que ofrece un rendimiento rápido y predecible, así como una perfecta escalabilidad. Para obtener más información sobre la transmisión de datos a OpenSearch Service, consulte the section called "Carga de datos de streaming desde Amazon DynamoDB".

Amazon Quick Suite

Puede visualizar los datos de OpenSearch Service mediante los paneles de Quick Suite. Para más información, consulte Uso de Amazon OpenSearch Service con Quick Suite en la Guía del usuario de Quick Suite.

> **Note**
>
> OpenSearch incluye código Elasticsearch con licencia de Apache de Elasticsearch B.V. y otro código fuente. Elasticsearch B.V. no es el origen de ese otro código fuente. ELASTICSEARCH es una marca comercial registrada de Elasticsearch B.V.

Configuración de Amazon OpenSearch Service

Concesión de permisos

En entornos de producción, le recomendamos que utilice políticas más específicas. Para obtener más información sobre la administración de acceso, consulte Recursos de AWS para administración de acceso en la Guía del usuario de IAM.

Para dar acceso, agregue permisos a los usuarios, grupos o roles:

- Usuarios y grupos en AWS IAM Identity Center:

  Cree un conjunto de permisos. Siga las instrucciones de Creación de un conjunto de permisos en la Guía del usuario de AWS IAM Identity Center.
- Usuarios gestionados en IAM a través de un proveedor de identidades:

  Cree un rol para la federación de identidades. Siga las instrucciones descritas en Creación de un rol para un proveedor de identidad de terceros (federación) en la Guía del usuario de IAM.
- Usuarios de IAM:
  - Cree un rol que el usuario pueda aceptar. Siga las instrucciones descritas en Creación de un rol para un usuario de IAM en la Guía del usuario de IAM.
  - (No recomendado) Adjunte una política directamente a un usuario o agregue un usuario a un grupo de usuarios. Siga las instrucciones descritas en Adición de permisos a un usuario (consola) de la Guía del usuario de IAM.

Concesión de acceso programático

Los usuarios necesitan acceso programático si desean interactuar con AWS fuera de la Consola de administración de AWS. La forma de conceder el acceso programático depende del tipo de usuario que acceda a AWS.

Para conceder acceso programático a los usuarios, seleccione una de las siguientes opciones.

¿Qué usuario necesita acceso programático?	Para	Mediante
Identidad del personal (Usuarios administrados en el IAM Identity Center)	Utiliza credenciales temporales para firmar las solicitudes programáticas a la AWS CLI, los AWS SDK y las API de AWS.	Siga las instrucciones de la interfaz que desea utilizar: • Para utilizar la AWS CLI, consulte Configuring the AWS CLI to usa AWS IAM Identity Center en la Guía del usuario de AWS Command Line Interface. • Para usar AWS SDK, las herramientas y las API de AWS, consulta IAM Identity Center authentication en la Guía de referencia del SDK y las herramientas de AWS.
IAM	Utiliza credenciales temporales para firmar las solicitudes programáticas a la AWS CLI, los AWS SDK y las API de AWS.	Siguiendo las instrucciones de Uso de credenciales temporales con recursos de AWS de la Guía del usuario de IAM.
IAM	(No recomendado) Utilizar credenciales a largo plazo para firmar las solicitudes programáticas a la AWS CLI, los AWS SDK o las API de AWS.	Siga las instrucciones de la interfaz que desea utilizar: • Para la AWS CLI, consulte Autenticación mediante credenciales de usuario de IAM en la Guía del usuario de AWS Command Line Interface. • Para ver los AWS SDK y las herramientas, consulte Autenticar mediante

¿Qué usuario necesita acceso programático?	Para	Mediante
		credenciales a largo plazo en la Guía de referencia de AWS SDK y herramientas. • Para las API de AWS, consulta Administración de claves de acceso para usuarios de IAM en la Guía del usuario de IAM.

Instalación y configuración de la AWS CLI

Si desea utilizar las API de OpenSearch Service, debe instalar la última versión de AWS Command Line Interface (AWS CLI). No necesita la AWS CLI para utilizar OpenSearch Service desde la consola y puede empezar sin la CLI siguiendo los pasos que se indican en Introducción a Amazon OpenSearch Service.

Configuración de la AWS CLI

1. Para instalar la versión más reciente de la AWS CLI para macOS, Linux o Windows, consulte Instalar o actualizar la última versión de la AWS CLI.
2. Para configurar la AWS CLI y asegurar la configuración de su acceso a los Servicios de AWS, incluido OpenSearch Service, consulte Configuración rápida con `aws configure`.
3. Para verificar la configuración, ingrese el siguiente comando DataBrew en el símbolo del sistema.

   ```
   aws opensearch help
   ```

   Los comandos de la AWS CLI utilizan la Región de AWS predeterminada de su configuración, a menos que lo configure con un parámetro o un perfil. Para configurar su Región de AWS con un parámetro, puede añadir el parámetro `--region` a cada comando.

   Para configurar su Región de AWS con un perfil, añada primero un perfil denominado en el archivo `~/.aws/config` o en el archivo `%UserProfile%/.aws/config` (para Microsoft

Windows). Siga los pasos en Perfiles denominados para la AWS CLI. A continuación, defina la Región de AWS y otros ajustes con un comando similar al del ejemplo siguiente.

```
[profile opensearch]
aws_access_key_id = ACCESS-KEY-ID-OF-IAM-USER
aws_secret_access_key = SECRET-ACCESS-KEY-ID-OF-IAM-USER
region = us-east-1
output = text
```

Abra la consola

La mayoría de los temas orientados a las consolas de esta sección comienzan desde la consola de OpenSearch Service. Si aún no ha iniciado sesión en su Cuenta de AWS, inicie sesión, abra la consola de OpenSearch Service y continúe con la siguiente sección para empezar a utilizar OpenSearch Service.

Introducción a Amazon OpenSearch Service

Para empezar, debe registrarse para obtener una cuenta de Cuenta de AWS si no dispone de una. Después de configurar la cuenta, complete el tutorial Introducción de Amazon OpenSearch Service. Consulte los siguientes temas introductorios si desea obtener más información y conocer el servicio:

- Crear un dominio.
- Ajuste el tamaño del dominio en función de la carga de trabajo.
- Controle el acceso a su dominio mediante una política de acceso al dominio o un control de acceso detallado.
- Indexe datos manualmente o desde otros servicios de AWS.
- Use OpenSearch Dashboards para buscar sus datos y crear visualizaciones.
- Conozca más opciones avanzadas para crear un dominio. Para más información, consulte Creación y administración de dominios.
- Descubra cómo administrar los índices de un dominio. Para más información, consulte Administración de índices.
- Pruebe uno de los tutoriales para trabajar con Amazon OpenSearch Service. Para obtener más información, consulte Tutoriales.

Para obtener información sobre la migración a OpenSearch Service desde un clúster autoadministrado de OpenSearch, consulte the section called “Migrar a OpenSearch Service”.

Para obtener información detallada, consulte Creación y administración de dominios y los demás temas de esta guía. Para obtener información sobre la migración a OpenSearch Service desde un clúster autoadministrado de OpenSearch, consulte the section called “Migrar a OpenSearch Service”.

Puede seguir estos pasos mediante la consola de OpenSearch Service, la AWS CLI o el AWS SDK. Para obtener más información sobre cómo instalar y configurar AWS CLI, consulte la Guía del usuario de AWS Command Line Interface.

Creación de un dominio de Amazon OpenSearch Service

> **Important**
>
> Este es un tutorial conciso para configurar una prueba de dominio de Amazon OpenSearch Service. No utilice este proceso para crear dominios de producción. Para ver una versión completa del mismo proceso, consulte Creación y administración de dominios.

Un dominio de OpenSearch Service es sinónimo de clúster de OpenSearch. Los dominios son clústeres con la configuración, los tipos de instancia, los recuentos de instancias y los recursos de almacenamiento que especifique. Puede crear un dominio de OpenSearch Service con la consola, la AWS CLI o los SDK de AWS.

Para crear un dominio de OpenSearch Service con la consola

1. Visite https://aws.amazon.com y elija Iniciar sesión en la consola.
2. En Análisis, seleccione Amazon OpenSearch Service.
3. Seleccione Crear un dominio.
4. Proporcione un nombre para el dominio. Los ejemplos de este tutorial utilizan el nombre Películas.
5. Seleccione Creación estándar para el método de creación del dominio.

   > **Note**
   >
   > Para configurar rápidamente un dominio de producción con las prácticas recomendadas, puede elegir Creación sencilla. Para el desarrollo y las pruebas de este tutorial, utilizaremos la Creación estándar.

6. Para las plantillas, seleccione Desarrollo/pruebas.
7. Para la opción de implementación, seleccione Dominio con modo de espera.
8. En Versión, seleccione la versión más reciente.
9. Por ahora, ignore las secciones de Nodos de datos, Almacenamiento de datos en caliente y en frío, Nodos maestros dedicados, Configuración de instantáneas y Puntos de conexión personalizados.

10. Para simplificar este tutorial, utilice un dominio de acceso público. En Red, seleccione Acceso público.
11. En la configuración de control de acceso detallado, mantenga seleccionada la casilla Habilitar el control de acceso detallado. Seleccione Crear usuario maestro e ingrese un nombre de usuario y una contraseña.
12. Por el momento, ignore las secciones Autenticación SAML y Autenticación de Amazon Cognito.
13. Para la Política de acceso, seleccione Utilizar únicamente control de acceso detallado. En este tutorial, el control de acceso detallado controla la autenticación, no la política de acceso al dominio.
14. Omita el resto de la configuración y elija Crear. Los dominios nuevos suelen tardar entre 15 y 30 minutos en inicializarse, pero pueden tardar más en función de la configuración. Una vez inicializado su dominio, selecciónelo para abrir su panel de configuración. Tome nota del punto de conexión del dominio en Información general (por ejemplo, `https://search-my-domain.us-east-1.es.amazonaws.com`), que deberá usar en el siguiente paso.

Siguiente: cargar datos en un dominio de OpenSearch Service para realizar la indexación

Carga de datos en Amazon OpenSearch Service para la indexación

> **Important**
>
> Este es un tutorial conciso para cargar una pequeña cantidad de datos de prueba en Amazon OpenSearch Service. Para obtener más información sobre cómo cargar datos en un dominio de producción, consulte Indexación de datos.

Puede cargar datos en un dominio de OpenSearch Service mediante la línea de comandos o la mayoría de los lenguajes de programación.

El siguiente ejemplo requiere utilizar curl, (un cliente HTTP común), para que el proceso sea más cómodo y rápido. Los clientes como curl no pueden realizar la firma de solicitudes necesaria si sus políticas de acceso especifican roles o usuarios de IAM. Para realizar correctamente este proceso, debe utilizar un control de acceso detallado con un nombre de usuario principal y una contraseña, como los configuró en el Paso 1.

Puede instalar curl en Windows y utilizarlo desde el símbolo del sistema, pero recomendamos utilizar una herramienta como Cygwin o Windows Subsystem for Linux. La distribución de macOS y la mayoría de las distribuciones de Linux vienen con curl preinstalado.

Opción 1: cargar un solo documento

Ejecute el siguiente comando para agregar un único documento al dominio Películas:

```
curl -XPUT -u 'master-user:master-user-password' 'domain-endpoint/movies/_doc/1' -d
 '{"director": "Burton, Tim", "genre": ["Comedy","Sci-Fi"], "year": 1996, "actor":
 ["Jack Nicholson","Pierce Brosnan","Sarah Jessica Parker"], "title": "Mars Attacks!"}'
 -H 'Content-Type: application/json'
```

En el comando, proporcione el nombre de usuario y la contraseña que creó en el Paso 1.

Para obtener una explicación detallada de este comando y cómo realizar solicitudes firmadas a OpenSearch Service, consulte Indexación de datos.

Opción 2: cargar varios documentos

Para cargar un archivo JSON que contiene varios documentos en un dominio de OpenSearch Service

1. Cree un archivo local denominado `bulk_movies.json`. Pegue el siguiente contenido en el archivo y agregue una línea nueva al final:

```
{ "index" : { "_index": "movies", "_id" : "2" } }
{"director": "Frankenheimer, John", "genre": ["Drama", "Mystery", "Thriller",
 "Crime"], "year": 1962, "actor": ["Lansbury, Angela", "Sinatra, Frank", "Leigh,
 Janet", "Harvey, Laurence", "Silva, Henry", "Frees, Paul", "Gregory, James",
 "Bissell, Whit", "McGiver, John", "Parrish, Leslie", "Edwards, James", "Flowers,
 Bess", "Dhiegh, Khigh", "Payne, Julie", "Kleeb, Helen", "Gray, Joe", "Nalder,
 Reggie", "Stevens, Bert", "Masters, Michael", "Lowell, Tom"], "title": "The
 Manchurian Candidate"}
{ "index" : { "_index": "movies", "_id" : "3" } }
{"director": "Baird, Stuart", "genre": ["Action", "Crime", "Thriller"], "year":
 1998, "actor": ["Downey Jr., Robert", "Jones, Tommy Lee", "Snipes, Wesley",
 "Pantoliano, Joe", "Jacob, Ir\u00e8ne", "Nelligan, Kate", "Roebuck, Daniel",
 "Malahide, Patrick", "Richardson, LaTanya", "Wood, Tom", "Kosik, Thomas",
 "Stellate, Nick", "Minkoff, Robert", "Brown, Spitfire", "Foster, Reese",
 "Spielbauer, Bruce", "Mukherji, Kevin", "Cray, Ed", "Fordham, David", "Jett,
 Charlie"], "title": "U.S. Marshals"}
```

```
{ "index" : { "_index": "movies", "_id" : "4" } }
{"director": "Ray, Nicholas", "genre": ["Drama", "Romance"], "year": 1955, "actor":
 ["Hopper, Dennis", "Wood, Natalie", "Dean, James", "Mineo, Sal", "Backus, Jim",
 "Platt, Edward", "Ray, Nicholas", "Hopper, William", "Allen, Corey", "Birch,
 Paul", "Hudson, Rochelle", "Doran, Ann", "Hicks, Chuck", "Leigh, Nelson",
 "Williams, Robert", "Wessel, Dick", "Bryar, Paul", "Sessions, Almira", "McMahon,
 David", "Peters Jr., House"], "title": "Rebel Without a Cause"}
```

2. Ejecute el siguiente comando en el directorio local en el que está almacenado el archivo para cargarlo en el dominio Películas:

```
curl -XPOST -u 'master-user:master-user-password' 'domain-endpoint/movies/_bulk' --
data-binary @bulk_movies.json -H 'Content-Type: application/x-ndjson'
```

Para obtener más información acerca del formato de archivo masivo, consulte Indexación de datos.

Siguiente: buscar documentos

Búsqueda de documentos en Amazon OpenSearch Service

Para buscar documentos en un dominio de Amazon OpenSearch Service, utilice la API de búsqueda de OpenSearch. También puede utilizar OpenSearch Dashboards para buscar documentos en el dominio.

Para buscar documentos desde la línea de comandos

Ejecute el siguiente comando para buscar el dominio Películas para la palabra mars:

```
curl -XGET -u 'master-user:master-user-password' 'domain-endpoint/movies/_search?
q=mars&pretty=true'
```

Si utilizó datos masivos en la página anterior, intente buscar rebel en su lugar.

Verá una respuesta parecida a la siguiente:

```
{
  "took" : 5,
  "timed_out" : false,
  "_shards" : {
```

```
    "total" : 5,
    "successful" : 5,
    "skipped" : 0,
    "failed" : 0
  },
  "hits" : {
    "total" : {
      "value" : 1,
      "relation" : "eq"
    },
    "max_score" : 0.2876821,
    "hits" : [
      {
        "_index" : "movies",
        "_type" : "_doc",
        "_id" : "1",
        "_score" : 0.2876821,
        "_source" : {
          "director" : "Burton, Tim",
          "genre" : [
            "Comedy",
            "Sci-Fi"
          ],
          "year" : 1996,
          "actor" : [
            "Jack Nicholson",
            "Pierce Brosnan",
            "Sarah Jessica Parker"
          ],
          "title" : "Mars Attacks!"
        }
      }
    ]
  }
}
```

Buscar documentos mediante OpenSearch Dashboards

OpenSearch Dashboards es una popular herramienta de visualización de código abierto diseñada para funcionar con OpenSearch. Proporciona una interfaz de usuario útil para que pueda buscar y monitorear índices.

Para buscar documentos de un dominio de OpenSearch Service mediante Dashboards

1. Desplácese hasta la URL de OpenSearch Dashboards para su dominio. Puede encontrar la URL en el panel del dominio en la consola de OpenSearch Service. La URL tiene este formato:

   ```
   domain-endpoint/_dashboards/
   ```

2. Inicie sesión con su nombre de usuario principal y contraseña.
3. Para utilizar Dashboards, debe crear al menos un patrón de índice. Dashboards utiliza estos patrones para identificar los índices que desea analizar. Abra el panel de navegación de la izquierda, seleccione Gestión de pilas, seleccione Patrones de índice y luego seleccione Crear patrón de índice. Para este tutorial, escriba Películas.
4. seleccione Siguiente paso y, a continuación, seleccione Crear patrón de índice. Una vez creado el patrón, puede ver los diversos campos de documento, como `actor` y `director`.
5. Regrese a la pestaña Patrones de índice y asegúrese de que `movies` se establezca como valor predeterminado. Si no lo es, seleccione el patrón y elija el icono de estrella para convertirlo en el predeterminado.
6. Para comenzar a buscar los datos, abra el panel de navegación izquierdo y elija Descubrir.
7. En la barra de búsqueda, escriba mars, si cargó un solo documento, o rebel, si cargó varios documentos y, a continuación, presione Enter. Puede buscar otros términos, como nombres de actores o directores.

Siguiente: eliminar un dominio

Eliminación de un dominio de Amazon OpenSearch Service

Dado que el dominio Películas de este tutorial es para hacer pruebas, debe eliminarlo cuando termine de experimentar para evitar incurrir en gastos.

Cómo eliminar un dominio de OpenSearch Service de la consola

1. Inicie sesión en la consola de Amazon OpenSearch Service.
2. En Dominios, seleccione el dominio películas.
3. Seleccione Eliminar y confirme la eliminación.

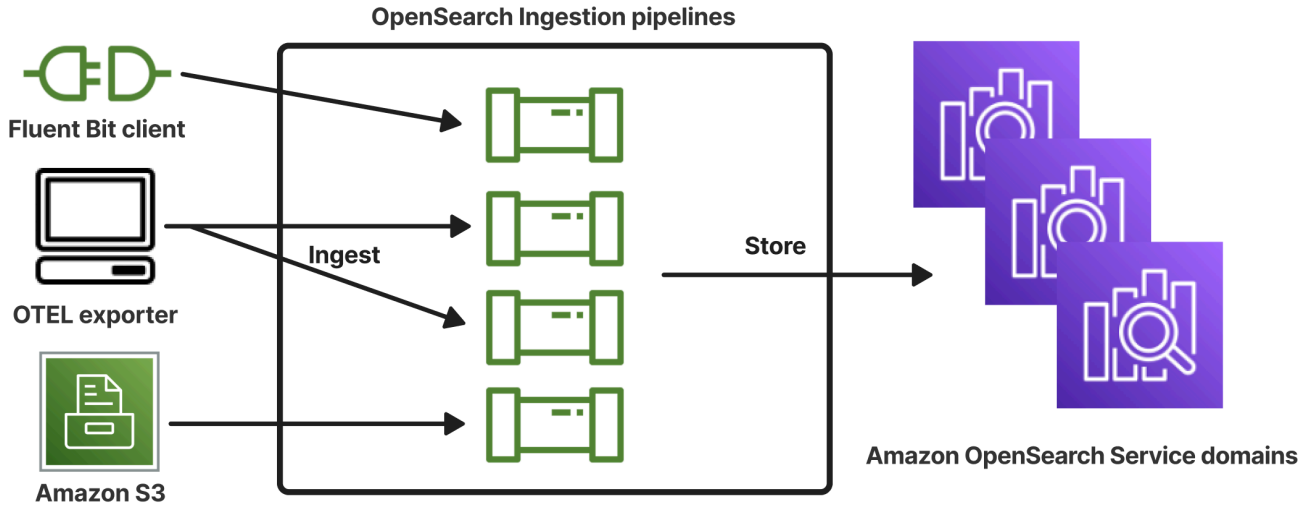
Descripción general de Amazon OpenSearch Ingestion

Amazon OpenSearch Ingestion es un recopilador de datos sin servidor, totalmente administrado, que transmite datos de registro, métricas y rastreo en tiempo real a los dominios de Amazon OpenSearch Service y a las colecciones de OpenSearch sin servidor.

Con OpenSearch Ingestion, ya no necesitará herramientas de terceros, como Logstash o Jaeger, para ingerir datos. Puede configurar los productores de datos para que envíen datos a OpenSearch Ingestion, que a su vez los envía automáticamente a la colección o dominio que especifique. También puede transformar los datos antes de entregarlos.

Como OpenSearch Ingestion funciona sin servidor, no tiene que administrar la infraestructura, parchear el software o escalar los clústeres manualmente. Puede aprovisionar los canales de ingesta directamente en la Consola de administración de AWS y OpenSearch Ingestion se encarga del resto.

Como componente de Amazon OpenSearch Service, OpenSearch Ingestion funciona con Data Prepper, un recopilador de datos de código abierto que filtra, enriquece, transforma, normaliza y agrega datos para su análisis y visualización posteriores.



Conceptos clave de Amazon OpenSearch Ingestion

Antes de empezar a utilizar OpenSearch Ingestion, es útil entender estos conceptos clave.

Canalización

Desde el punto de vista de OpenSearch Ingestion, una canalización hace referencia a un único recopilador de datos aprovisionado que se crea en OpenSearch Service. Puede considerarlo

como el archivo de configuración de YAML completo, que incluye una o más subcanalizaciones. Para ver los pasos para crear una canalización de ingesta, consulte the section called “Creación de canalizaciones”.

Subcanalización

Las subcanalizaciones se definen dentro de un archivo de configuración YAML. Cada subcanalización es una combinación de una sola fuente, un búfer, cero o más procesadores y uno o más receptores. Puede definir varias subcanalizaciones en un único archivo YAML, cada una con fuentes, procesadores y receptores únicos. Para facilitar la supervisión con CloudWatch y otros servicios, le recomendamos que especifique un nombre de canalización que sea distinto de todas sus subcanalizaciones.

Puede encadenar varias subcanalizaciones en un único archivo YAML, de modo que la fuente de una subcanalización sea otra subcanalización y su receptor sea una tercera subcanalización. Para ver un ejemplo, consulta the section called “Recolector de OpenTelemetry”.

Origen

El componente de entrada de una subcanalización. Define el mecanismo mediante el cual una canalización consume los registros. La fuente puede consumir eventos ya sea recibiéndolos a través de HTTPS o leyendo desde puntos de conexión externos, como Amazon S3. Hay dos tipos de fuentes: basadas en push y basadas en pull. Las fuentes basadas en push, como los registros HTTP y OTel, transmiten los registros a los puntos de conexión de ingesta. Las fuentes basadas en pull, como el rastreo OTel y S3, extraen datos de la fuente.

Procesadores

Unidades de procesamiento intermedias que pueden filtrar, transformar y enriquecer los registros en el formato deseado antes de publicarlos en el receptor. El procesador es un componente opcional de una canalización. Si no define un procesador, los registros se publican en el formato definido en la fuente. Puede tener más de un procesador. Una canalización ejecuta los procesadores en el orden en el que los define.

Receptor

El componente de salida de una subcanalización. Define uno o más destinos en los que una subcanalización publica los registros. OpenSearch Ingestion admite los dominios de OpenSearch Service como receptores. También admite subcanalizaciones como receptores. Esto significa que puede encadenar varias subcanalizaciones en una sola canalización de OpenSearch Ingestion (archivo YAML). Los clústeres de OpenSearch autoadministrados no se admiten como receptores.

Búfer

La parte de un procesador que actúa como capa entre la fuente y el receptor. No puede configurar manualmente un búfer dentro de la canalización. OpenSearch Ingestion usa una configuración de búfer predeterminada.

Ruta

La parte de un procesador que permite a los autores de canalizaciones enviar únicamente los eventos que cumplan determinadas condiciones a distintos receptores.

Una definición de subcanalización válida debe contener una fuente y un receptor. Para obtener más información sobre cada uno de estos elementos de la canalización, consulte Referencia de configuración.

Beneficios de Amazon OpenSearch Ingestion

OpenSearch Ingestion tiene los siguientes beneficios principales:

- Elimina la necesidad de administrar manualmente una canalización autoaprovisionada.
- Escala automáticamente las canalizaciones en función de los límites de capacidad que defina.
- Mantiene la canalización actualizada con parches de seguridad y de errores.
- Ofrece la opción de conectar las canalizaciones a su nube privada virtual (VPC) para añadir un nivel de seguridad adicional.
- Le permite detener e iniciar canalizaciones para controlar los costos.
- Proporciona esquemas de configuración de canalizaciones para los casos de uso más habituales a fin de ayudarle a empezar a trabajar más rápido.
- Le permite interactuar mediante programación con las canalizaciones a través de los distintos SDK de AWS y la API de OpenSearch Ingestion.
- Admite la supervisión del rendimiento en Amazon CloudWatch y el registro de errores en Registros de CloudWatch.

Limitaciones de Amazon OpenSearch Ingestion

OpenSearch Ingestion tiene las siguientes limitaciones:

- Solo puede incorporar datos en dominios que ejecuten OpenSearch 1.0 o posterior o Elasticsearch 6.8 o posterior. Si utiliza la fuente de rastreo OTel, le recomendamos que utilice Elasticsearch 7.9 o una versión posterior para poder usar el complemento de OpenSearch Dashboards.
- Si una canalización se escribe en un dominio de OpenSearch Service que se encuentra dentro de una VPC, la canalización debe crearse en la misma Región de AWS que el dominio.
- Solo puede configurar un único origen de datos dentro de una definición de canalización.
- No puede especificar clústeres de OpenSearch autoadministrados como receptores.
- No puede especificar un punto de conexión personalizado como receptor. Aún puede escribir en un dominio que tenga habilitados los puntos de conexión personalizados, pero debes especificar su punto de conexión estándar.
- No puede especificar los recursos dentro de las regiones registradas como orígenes o receptores.
- Existen algunas limitaciones en los parámetros que se pueden incluir en una configuración de canalización. Para más información, consulte the section called “Requisitos y restricciones de configuración”.

Versiones de Data Prepper admitidas

Actualmente, OpenSearch Ingestion es compatible con las siguientes versiones principales de Data Prepper:

- 2.x

Al crear una canalización con el Editor de código, utilice la opción de `version` necesaria para especificar la versión principal de Data Prepper que va a utilizar. Por ejemplo, `version: "2"`. OpenSearch Ingestion recupera la última versión secundaria compatible de esa versión principal y aprovisiona la canalización con esa versión.

Si no utiliza el editor de código para crear su canalización, OpenSearch Ingestion aprovisiona automáticamente su canalización con la última versión compatible.

Actualmente, las canalizaciones de OpenSearch Ingestion se aprovisionan con la versión 2.7 de Data Prepper. Para obtener información, consulte las notas de la versión 2.7. OpenSearch Ingestion no admite todas las versiones secundarias de una versión principal en particular.

Al actualizar la configuración de una canalización, si se admite una nueva versión secundaria de Data Prepper, OpenSearch Ingestion actualiza automáticamente la canalización a la última versión

secundaria compatible de la versión principal especificada en la configuración de la canalización. Por ejemplo, es posible que tenga `version: "2"` en la configuración de su canalización y OpenSearch Ingestion aprovisionó inicialmente la canalización con la versión 2.6.0. Cuando se agrega la compatibilidad con la versión 2.7.0 y hace un cambio en la configuración de la canalización, OpenSearch Ingestion la actualiza a la versión 2.7.0. Este proceso mantiene su canalización actualizada con las últimas correcciones de errores y mejoras de rendimiento. OpenSearch Ingestion no puede actualizar la versión principal de su canalización a menos que cambie manualmente la opción `version`en la configuración de la canalización. Para obtener más información, consulte the section called “Actualización de las canalizaciones”.

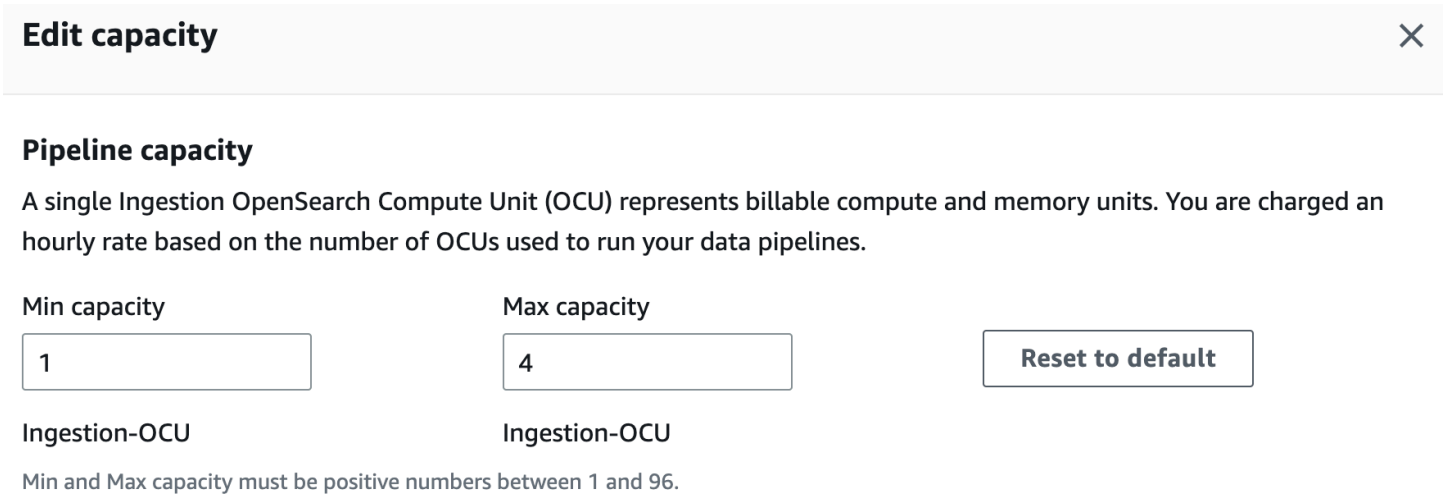
Escalado de canalizaciones en Amazon OpenSearch Ingestion

OpenSearch Ingestion escala automáticamente la capacidad de las canalizaciones en función de las unidades de cómputo OpenSearch de ingesta (OCU de ingesta) mínimas y máximas que especifique. Esto elimina la necesidad de aprovisionamiento y administración manuales.

Cada OCU de ingesta es una combinación de aproximadamente 15 GiB de memoria y 2 vCPU. Puede especificar los valores de OCU mínimos y máximos para una canalización, y OpenSearch Ingestion escala automáticamente la capacidad de la canalización en función de esos límites.

Especifique los siguientes valores al crear una canalización:

- Capacidad mínima: la canalización puede reducir la capacidad hasta este número de OCU de ingesta. La capacidad mínima especificada también es la capacidad inicial de una canalización.
- Capacidad máxima: la canalización puede aumentar la capacidad hasta este número de OCU de ingesta.



Asegúrese de que la capacidad máxima de la canalización sea lo suficientemente alta como para administrar los picos de carga de trabajo y que la capacidad mínima sea lo suficientemente baja como para minimizar los costos cuando la canalización no esté ocupada. Según su configuración, OpenSearch Ingestion escala de forma automática la cantidad de OCU de ingesta de sus canalizaciones para procesar la carga de trabajo de ingesta. En cualquier momento dado, solo se le cobrarán las OCU de ingesta que se estén utilizando activamente en su canalización.

La capacidad asignada a su canalización de OpenSearch Ingestion se escala y se reduce verticalmente en función de los requisitos de procesamiento de la canalización y la carga generada por la aplicación cliente. Cuando la capacidad es limitada, OpenSearch Ingestion escala verticalmente mediante la asignación de más unidades de cómputo (GiB de memoria). Cuando la canalización procese cargas de trabajo más pequeñas o no procese datos en absoluto, puede reducir verticalmente hasta el mínimo de OCU de ingesta configuradas.

Puede especificar un mínimo de 1 OCU de ingesta, un máximo de 96 OCU de ingesta para las canalizaciones sin estado y un máximo de 48 OCU de ingesta para las canalizaciones con estado. Recomendamos un mínimo de 2 OCU de ingesta como mínimo para las fuentes push. Cuando el almacenamiento en búfer persistente esté activado, puede especificar un mínimo de 2 y un máximo de 384 OCU de ingesta.

Dada una canalización de registro estándar con una sola fuente, un patrón grok simple y un receptor, cada unidad de cómputo puede admitir hasta 2 MiB por segundo. En el caso de las canalizaciones de registro más complejas con varios procesadores, cada unidad de cómputo puede soportar una carga de ingesta menor. En función de la capacidad de la canalización y el uso de los recursos, se inicia el proceso de escalado de OpenSearch Ingestion.

Para garantizar una alta disponibilidad, las OCU de ingesta se distribuyen en las zonas de disponibilidad (AZ). La cantidad de AZ depende de la capacidad mínima que se especifique.

Por ejemplo, si especifica un mínimo de 2 unidades de cómputo, las OCU de ingesta que estén en uso en un momento dado se distribuyen uniformemente entre las 2 AZ. Si especifica un mínimo de 3 o más unidades de cómputo, las OCU de ingesta se distribuyen uniformemente en 3 AZ. Le recomendamos que aprovisione al menos dos OCU de ingesta para garantizar una disponibilidad del 99,9 % en sus canalizaciones de ingesta.

No se le facturan las OCU de ingesta cuando una canalización está en los estados `Create failed`, `Creating`, `Deleting` y `Stopped`.

Para ver instrucciones sobre cómo configurar y recuperar los ajustes de capacidad de una canalización, consulte the section called “Creación de canalizaciones”.

Precios de OpenSearch Ingestion

En un momento específico, solo paga por la cantidad de OCU de ingesta que estén asignadas a una canalización, independientemente de si hay datos circulando por la canalización. OpenSearch Ingestion se adapta inmediatamente a sus cargas de trabajo al escalar o reducir la capacidad de la canalización en función del uso.

Para obtener información completa sobre precios, consulte los precios de Amazon OpenSearch Service.

Regiones de AWS compatible

OpenSearch Ingestion está disponible en un subconjunto de Regiones de AWS en las que está disponible OpenSearch Service. Para obtener una lista de las regiones compatibles, consulte Puntos de conexión y cuotas de Amazon OpenSearch Service en Referencia general de AWS.

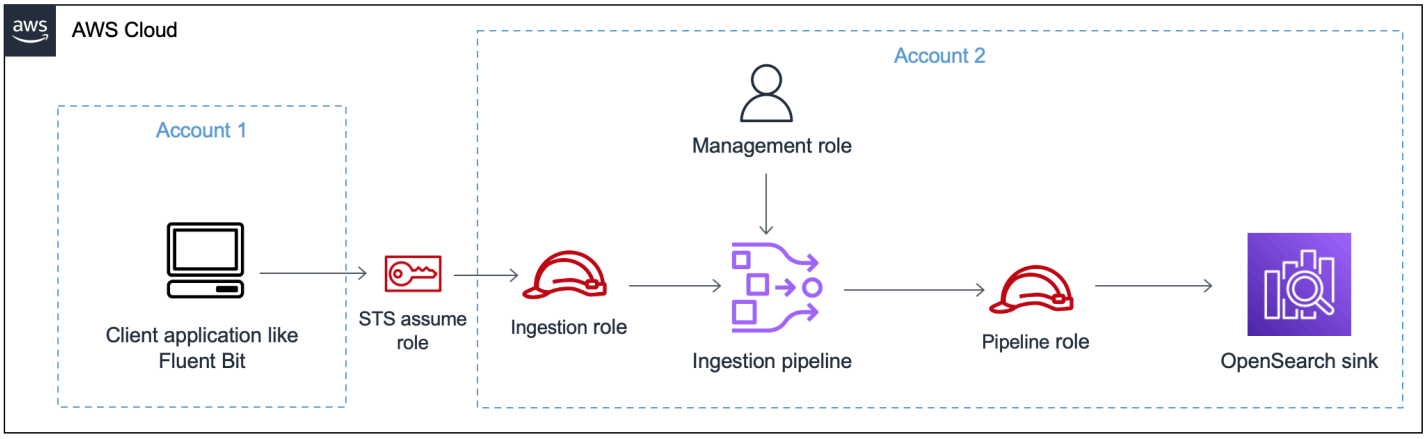
Configuración de roles y usuarios en Amazon OpenSearch Ingestion

Amazon OpenSearch Ingestion utiliza diversos modelos de permisos y roles de IAM para permitir que las aplicaciones de origen escriban en las canalizaciones y que las canalizaciones escriban en los receptores. Antes de empezar a incorporar datos, debe crear uno o más roles de IAM con permisos específicos en función de su caso de uso.

Como mínimo, se requieren los siguientes roles para configurar una canalización exitosa.

Nombre	Descripción
Rol de canalización	El rol de canalización proporciona los permisos necesarios para que una canalización lea desde el origen y escriba en el receptor del dominio o de la colección. Puede crear el rol de canalización manualmente o puede hacer que OpenSearch Ingestion lo cree por usted.
Rol de ingesta	El rol de ingestión contiene el permiso `osis:Ingest` para el recurso de canalización. Este permiso permite que los orígenes basados en push incorporen datos en una canalización.

La siguiente imagen muestra una configuración de canalización típica, en la que un origen de datos como Amazon S3 o Fluent Bit escribe en una canalización de una cuenta diferente. En este caso, el cliente debe asumir el rol de incorporación para poder acceder a la canalización. Para obtener más información, consulte the section called “Incorporación entre cuentas”.



Para obtener una guía de configuración sencilla, consulte the section called “Tutorial: incorporar datos a un dominio”.

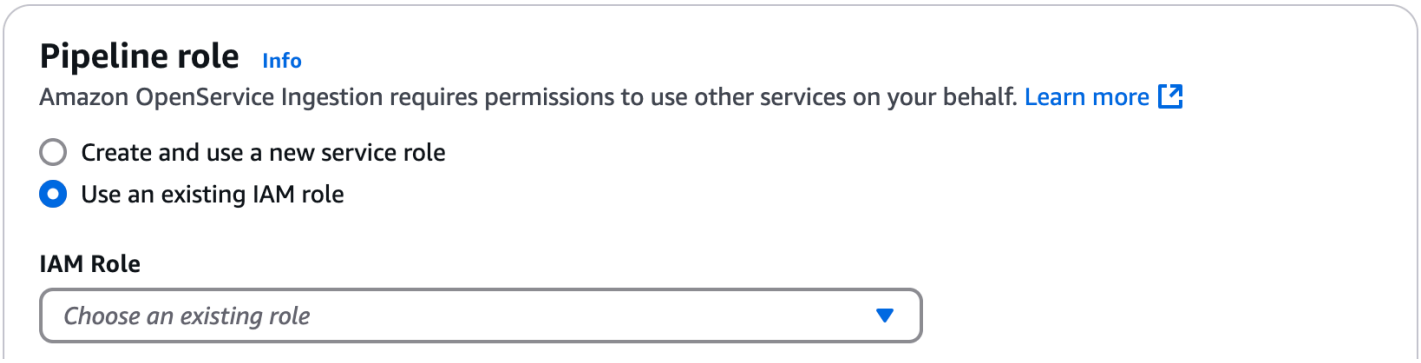
Temas

- the section called “Rol de canalización”
- the section called “Rol de ingesta”
- the section called “Incorporación entre cuentas”

Rol de canalización

Una canalización necesita ciertos permisos para leer de su fuente y escribir en su receptor. Estos permisos dependen de la aplicación cliente o del Servicio de AWS que esté escribiendo en la canalización y de si el receptor es un dominio de OpenSearch Service, una colección de OpenSearch sin servidor o Amazon S3. Además, es posible que una canalización necesite permisos para extraer información de la aplicación de origen (si el origen es un complemento basado en la extracción) y permisos para escribir en una cola de mensajes fallidos de S3, si está configurada.

Al crear una canalización, tiene la opción de especificar un rol de IAM existente que haya creado manualmente o hacer que OpenSearch Ingestion cree automáticamente el rol de canalización en función del origen y el receptor que haya seleccionado. La siguiente imagen muestra cómo especificar el rol de canalización en la Consola de administración de AWS.



Temas

- Automatización de la creación de roles de canalización
- Creación manual del rol de canalización

Automatización de la creación de roles de canalización

Puede elegir que OpenSearch Ingestion cree el rol de canalización por usted. Identifica automáticamente los permisos que requiere el rol en función del origen y los receptores configurados. Crea un rol de IAM con el prefijo `OpenSearchIngestion-` y el sufijo que introduzcas. Por ejemplo, si introduce `PipelineRole` como sufijo, OpenSearch Ingestion crea un rol denominado `OpenSearchIngestion-PipelineRole`.

La creación automática del rol de canalización simplifica el proceso de configuración y reduce la probabilidad de que se produzcan errores de configuración. Al automatizar la creación de roles, puede evitar la asignación manual de permisos y garantizar que se apliquen las políticas correctas

sin correr el riesgo de errores de configuración de seguridad. Esto también ahorra tiempo y mejora el cumplimiento de las normas de seguridad al aplicar las mejores prácticas y, al mismo tiempo, garantizar la coherencia en las distintas implementaciones en proceso.

Solo puede hacer que OpenSearch Ingestion cree automáticamente el rol de canalización en la Consola de administración de AWS. Si utiliza la AWS CLI, la API de OpenSearch Ingestion o uno de los SDK, debe especificar un rol de canalización creado manualmente.

Elija Crear y utilizar un nuevo rol de servicio para que OpenSearch Ingestion cree el rol por usted.

> **Important**
>
> Aún tiene que modificar manualmente la política de acceso al dominio o a la colección para conceder el acceso a la función de canalización. En el caso de los dominios que utilicen un control de acceso detallado, también debe asignar el rol de canalización a un rol de backend. Puede realizar estos pasos antes o después de crear la canalización.
> Para obtener instrucciones, consulte los siguientes temas:
>
> - Configurar el acceso a los datos para el dominio
> - Configurar el acceso a los datos y a la red para la colección

Creación manual del rol de canalización

Es posible que prefiera crear el rol de canalización de forma manual si necesita tener más control sobre los permisos para cumplir requisitos específicos de seguridad o conformidad. La creación manual le permite personalizar los roles para adaptarlos a las estrategias de administración de acceso o de infraestructura existentes. También puede optar por la configuración manual para integrar el rol con otros Servicios de AWS o asegurarse de que se ajuste a sus necesidades operativas únicas.

Para elegir un rol de canalización creado manualmente, seleccione Usar un rol de IAM existente y elija un rol existente. El rol debe tener todos los permisos necesarios para recibir datos del origen seleccionado y escribir en el receptor seleccionado. En las siguientes secciones se describe cómo crear manualmente un rol de canalización.

Temas

- Permisos para leer desde un origen

- Permisos para escribir en un receptor de dominio
- Permisos para escribir en un receptor de colecciones
- Permisos para escribir en Amazon S3 o en una cola de mensajes fallidos

Permisos para leer desde un origen

Una canalización de OpenSearch Ingestion necesita permiso para leer y recibir datos del origen especificado. Por ejemplo, para un origen de Amazon DynamoDB, necesita permisos como `dynamodb:DescribeTable` y `dynamodb:DescribeStream`. Para ver ejemplos de políticas de acceso a roles de canalización para orígenes comunes, como Amazon S3, Fluent Bit y OpenTelemetry Collector, consulte the section called “Integración de canalizaciones”.

Permisos para escribir en un receptor de dominio

Una canalización de OpenSearch Ingestion necesita permiso para escribir en un dominio de OpenSearch Service que esté configurado como su receptor. Estos permisos incluyen la capacidad de describir el dominio y enviarle solicitudes HTTP. Estos permisos son los mismos para los dominios públicos y de VPC. Para obtener instrucciones acerca de cómo crear un rol de canalización y especificarlo en la política de acceso a dominios, consulte Permitir que las canalizaciones accedan a los dominios.

Permisos para escribir en un receptor de colecciones

Una canalización de OpenSearch Ingestion necesita permiso para escribir en una colección de OpenSearch sin servidor que esté configurada como su receptor. Estos permisos incluyen la capacidad de describir la colección y enviarle solicitudes HTTP.

En primer lugar, asegúrese de que su política de acceso a los roles de canalización conceda los permisos necesarios. A continuación, incluya este rol en una política de acceso a los datos y asígnele permisos para crear índices, actualizar índices, describir índices y escribir documentos dentro de la colección. Para obtener instrucciones para completar cada uno de estos pasos, consulte Allowing pipelines to access collections.

Permisos para escribir en Amazon S3 o en una cola de mensajes fallidos

Si especifica Amazon S3 como destino receptor de su canalización, o si habilita una cola de mensajes fallidos (DLQ), el rol de canalización debe permitirle acceder al bucket de S3 que especifique como destino.

Adjunte una política de permisos independiente al rol de canalización que otorgue acceso a la DLQ. Como mínimo, debe otorgarse la acción `S3:PutObject` al rol en el recurso del bucket:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Sid": "WriteToS3DLQ",
      "Effect": "Allow",
      "Action": "s3:PutObject",
      "Resource": "arn:aws:s3:::my-dlq-bucket/*"
    }
  ]
}
```

Rol de ingesta

El rol de ingesta es un rol de IAM que permite a los servicios externos interactuar de forma segura con una canalización de OpenSearch Ingestion y enviar datos a ella. En el caso de los orígenes basados en push, como Amazon Security Lake, este rol debe conceder permisos para introducir datos en la canalización, incluido `osis:Ingest`. En el caso de los orígenes basados en extracciones, como Amazon S3, el rol debe permitir que OpenSearch Ingestion lo asuma y acceda a los datos con los permisos necesarios.

Temas

- Rol de ingesta para orígenes basados en push
- Rol de ingesta de orígenes basados en extracciones
- Incorporación entre cuentas

Rol de ingesta para orígenes basados en push

En el caso de los orígenes basados en push, los datos se envían o insertan en la canalización de ingesta desde otro servicio, como Amazon Security Lake o Amazon DynamoDB. En este escenario, el rol de ingesta necesita, como mínimo, el permiso `osis:Ingest` para interactuar con la canalización.

La siguiente política de acceso de IAM demuestra cómo conceder este permiso al rol de ingesta:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "osis:Ingest"
      ],
      "Resource": "arn:aws:osis:us-east-1:111122223333:pipeline/pipeline-name/*"
    }
  ]
}
```

Rol de ingesta de orígenes basados en extracciones

En el caso de los orígenes basados en extracciones, la canalización de OpenSearch Ingestion extrae o recupera de forma activa datos de un origen externo, como Amazon S3. En este caso, la canalización debe asumir un rol de canalización de IAM que conceda los permisos necesarios para acceder al origen de datos. En estos escenarios, el rol de ingesta es sinónimo del rol de canalización.

El rol debe incluir una relación de confianza que permita a OpenSearch Ingestion asumirla y permisos específicos de origen de datos. Para obtener más información, consulte the section called “Permisos para leer desde un origen”.

Incorporación entre cuentas

Es posible que tenga que incorporar datos a una canalización desde una Cuenta de AWS diferente, como una cuenta de aplicación. Para configurar la incorporación entre cuentas, defina un rol de incorporaco+pm dentro de la misma cuenta que la canalización y establezca una relación de confianza entre el rol de ingesta y la cuenta de la aplicación:

JSON

```
{
  "Version":"2012-10-17",
```

```
  "Statement": [{
     "Effect": "Allow",
     "Principal": {
       "AWS": "arn:aws:iam::444455556666:root"
      },
     "Action": "sts:AssumeRole"
  }]
}
```

A continuación, configure la aplicación para que asuma el rol de incorporación. La cuenta de la aplicación debe conceder al rol de aplicación permisos AssumeRole para el rol de incorporación en la cuenta de canalización.

Para ver ejemplos de pasos detallados y ejemplos de políticas de IAM, consulte the section called "Provisión de acceso de ingesta entre cuentas".

Concesión de acceso a dominios para canalizaciones de Amazon OpenSearch Ingestion

Una canalización de Amazon OpenSearch Ingestion necesita permiso para escribir en el dominio de OpenSearch Service que esté configurado como su receptor. Para otorgar acceso, debe configurar un rol de AWS Identity and Access Management (IAM) con una política de permisos restrictiva que limite el acceso al dominio al que una canalización envía datos. Por ejemplo, es posible que desee limitar una canalización de incorporación únicamente al dominio y los índices necesarios para respaldar su caso de uso.

> **Important**
>
> Puede optar por crear el rol de canalización manualmente o puede hacer que OpenSearch Ingestion lo cree por usted durante la creación del canalización. Si elige la creación automática de roles, OpenSearch Ingestion agregará todos los permisos necesarios a la política de acceso a los roles en proceso en función del origen y el receptor que seleccione. Crea un rol de canalización en IAM con el prefijo `OpenSearchIngestion-` y el sufijo que introduzca. Para obtener más información, consulte the section called "Rol de canalización". Si ha pedido a OpenSearch Ingestion que cree el rol de canalización por usted, tendrá que incluirlo en la política de acceso al dominio y asignarlo a un rol de backend (si el dominio utiliza un control de acceso preciso), ya sea antes o después de crear la canalización. Para obtener instrucciones, consulte el paso 2.

Temas

- Paso 1: crear el rol de canalización
- Paso 2: Configurar el acceso a los datos del dominio

Paso 1: crear el rol de canalización

El rol debe tener una política de permisos adjunta que le permita enviar datos al receptor del dominio. También debe tener una relación de confianza que permita a OpenSearch Ingestion asumir el rol. Para ver instrucciones sobre cómo adjuntar una política a un rol, consulte Adición de permisos de identidad de IAM en la Guía del usuario de IAM.

En el siguiente ejemplo de política, se muestra el privilegio mínimo que se puede proporcionar en el rol de una canalización para escribir en un solo dominio:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "es:DescribeDomain",
            "Resource": "arn:aws:es:*:111122223333:domain/*"
        },
        {
            "Effect": "Allow",
            "Action": "es:ESHttp*",
            "Resource": "arn:aws:es:*:111122223333:domain/domain-name/*"
        }
    ]
}
```

Si piensa reutilizar el rol para escribir en varios dominios, puede ampliar la política sustituyendo el nombre de dominio por un carácter comodín (*).

El rol debe tener la siguiente relación de confianza, lo que permite a OpenSearch Ingestion asumir el rol de canalización:

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Principal":{
            "Service":"osis-pipelines.amazonaws.com"
         },
         "Action":"sts:AssumeRole"
      }
   ]
}
```

Paso 2: Configurar el acceso a los datos del dominio

Para que una canalización escriba datos en un dominio, el dominio debe tener una política de acceso de nivel de dominio que permita al rol de canalización acceder a ellos.

El siguiente ejemplo de política de acceso al dominio permite que el rol de canalización denominado `pipeline-role` escriba datos en el dominio denominado `ingestion-domain`:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:role/pipeline-role"
            },
            "Action": [
                "es:DescribeDomain",
                "es:ESHttp*"
            ],
            "Resource": "arn:aws:es:us-east-1:111122223333:domain/domain-name/*"
        }
    ]
```

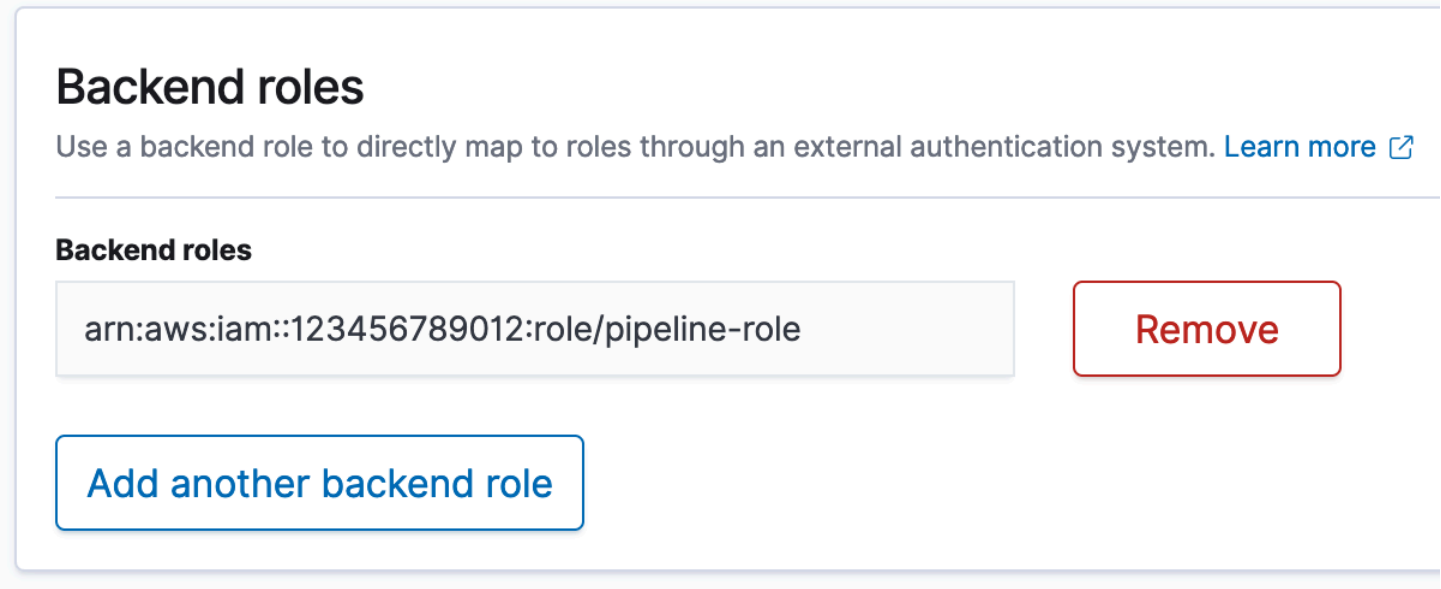
```
}
```

Asignar el rol de canalización (solo para dominios que utilicen un control de acceso detallado)

Si su dominio utiliza un control de acceso detallado para la autenticación, debe seguir algunos pasos adicionales para proporcionar a su canalización acceso a un dominio. Los pasos varían en función de la configuración del dominio:

- Escenario 1: rol de maestro y rol de canalización diferentes: si utiliza un nombre de recurso de Amazon (ARN) de IAM como usuario maestro y es diferente del rol de canalización, debe asignar el rol de canalización al rol de backend de OpenSearch `all_access`. Esto agrega el rol de canalización como usuario maestro adicional. Para obtener más información, consulte Usuarios maestros adicionales.
- Escenario 2: usuario maestro en la base de datos de usuarios interna: si su dominio utiliza un usuario maestro en la base de datos de usuarios interna y la autenticación básica HTTP para OpenSearch Dashboards, no podrá pasar el nombre de usuario y la contraseña maestros directamente a la configuración de la canalización. En su lugar, debe asignar el rol de canalización al rol de backend de OpenSearch `all_access`. Esto agrega el rol de canalización como usuario maestro adicional. Para obtener más información, consulte Usuarios maestros adicionales.
- Escenario 3: el mismo rol de maestro y rol de canalización (poco común): si utiliza un ARN de IAM como usuario maestro y es el mismo ARN que utiliza como rol de canalización, no necesita realizar ninguna otra acción. La canalización tiene los permisos necesarios para escribir en el dominio. Este escenario es poco común porque la mayoría de los entornos utilizan un rol de administrador o algún otro rol como rol de maestro.

La siguiente imagen muestra cómo asignar el rol de canalización a un rol de backend:



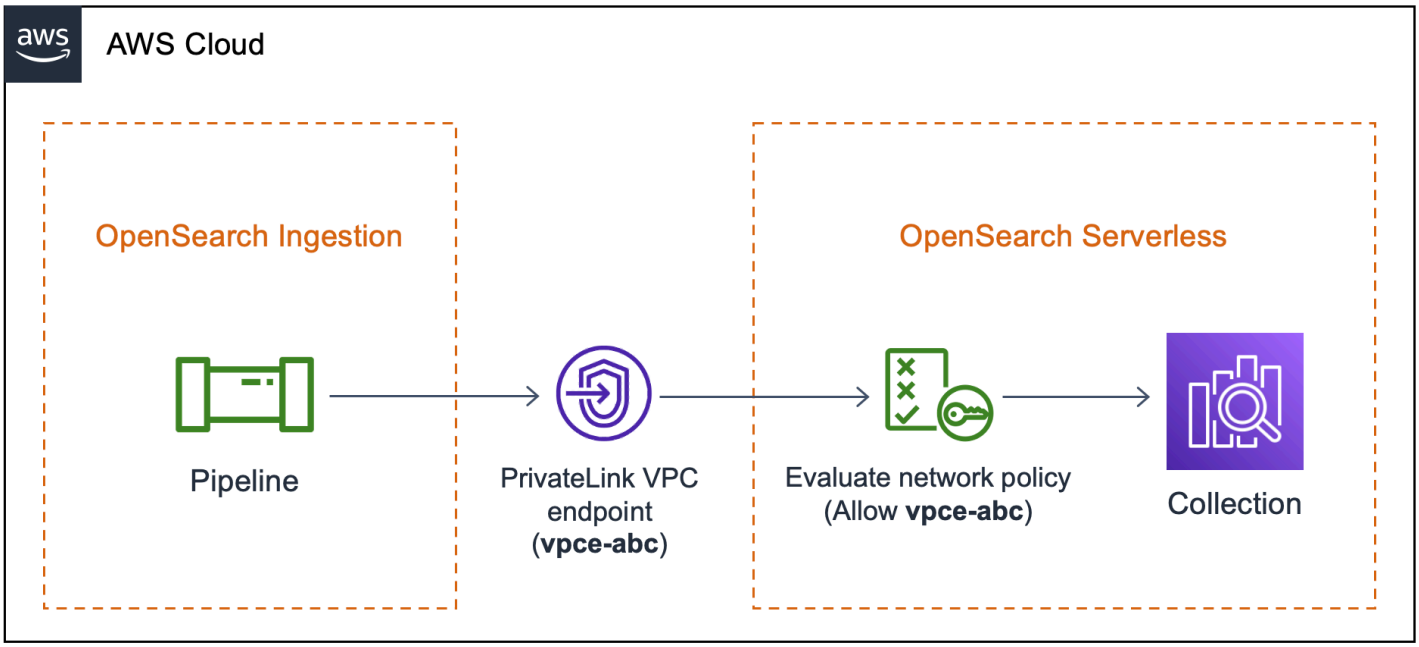
Concesión de acceso a colecciones para canalizaciones de Amazon OpenSearch Ingestion

Una canalización de Amazon OpenSearch Ingestion puede escribir en una colección pública o en una colección de VPC de OpenSearch sin servidor. Para proporcionar acceso a la colección, debe configurar un rol de canalización de AWS Identity and Access Management (IAM) con una política de permisos que conceda el acceso a la colección. La canalización asume este rol para firmar las solicitudes para el receptor de colecciones de OpenSearch sin servidor.

> **Important**
>
> Puede optar por crear el rol de canalización manualmente o puede hacer que OpenSearch Ingestion lo cree por usted durante la creación del canalización. Si elige la creación automática de roles, OpenSearch Ingestion agregará todos los permisos necesarios a la política de acceso a los roles en proceso en función del origen y el receptor que seleccione. Crea un rol de canalización en IAM con el prefijo `OpenSearchIngestion-` y el sufijo que introduzca. Para obtener más información, consulte the section called “Rol de canalización”. Si ha pedido a OpenSearch Ingestion que cree la función de canalización por usted, tendrá que incluir la función en la política de acceso a los datos de la recopilación, antes o después de crear la canalización. Para obtener instrucciones, consulte el paso 2.

Durante la creación de la canalización, OpenSearch Ingestion crea una conexión de AWS PrivateLink entre la canalización y la colección de OpenSearch sin servidor. Todo el tráfico de la canalización

pasa por este punto de conexión de VPC y se enruta a la colección. Para llegar a la colección, el punto de conexión debe tener acceso a la colección mediante una política de acceso a la red.



Temas

- Paso 1: crear el rol de canalización
- Paso 2: Configurar el acceso a los datos y a la red para la colección

Paso 1: crear el rol de canalización

El rol debe tener una política de permisos adjunta que le permita enviar datos al receptor de la colección. También debe tener una relación de confianza que permita a OpenSearch Ingestion asumir el rol. Para ver instrucciones sobre cómo adjuntar una política a un rol, consulte Adición de permisos de identidad de IAM en la Guía del usuario de IAM.

En el siguiente ejemplo de política, se muestra el privilegio mínimo que se puede proporcionar en la política de acceso al rol de una canalización para escribir en las colecciones:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
```

```
            "Sid": "Statement1",
            "Effect": "Allow",
            "Action": [
                "aoss:APIAccessAll",
                "aoss:BatchGetCollection",
                "aoss:CreateSecurityPolicy",
                "aoss:GetSecurityPolicy",
                "aoss:UpdateSecurityPolicy"
            ],
            "Resource": "*"
        }
    ]
}
```

El rol debe tener la siguiente relación de confianza, que permita a OpenSearch Ingestion asumirlo:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "osis-pipelines.amazonaws.com"
            },
            "Action": "sts:AssumeRole"
        }
    ]
}
```

Paso 2: Configurar el acceso a los datos y a la red para la colección

Cree una colección OpenSearch sin servidor con la siguiente configuración. Para ver instrucciones sobre cómo crear una colección, consulte the section called “Creación de colecciones”.

Política de acceso a los datos

Cree una política de acceso a los datos para la colección que conceda los permisos necesarios para el rol de canalización. Por ejemplo:

```
[
  {
    "Rules": [
      {
        "Resource": [
          "index/collection-name/*"
        ],
        "Permission": [
          "aoss:CreateIndex",
          "aoss:UpdateIndex",
          "aoss:DescribeIndex",
          "aoss:WriteDocument"
        ],
        "ResourceType": "index"
      }
    ],
    "Principal": [
      "arn:aws:iam::account-id:role/pipeline-role"
    ],
    "Description": "Pipeline role access"
  }
]
```

> **Note**
>
> En el elemento `Principal`, debe especificar el nombre de recurso de Amazon (ARN) de un rol de canalización.

Política de acceso a la red

Cada colección que cree en OpenSearch sin servidor tiene asociada al menos una política de acceso a la red. Las políticas de acceso a la red determinan si se puede acceder a la colección a través de Internet desde redes públicas o si se debe acceder a ella de forma privada. Para obtener más información sobre las políticas de red, consulte the section called “Acceso a la red”.

En una política de acceso a la red, solo puede especificar puntos de conexión de VPC administrados por OpenSearch sin servidor. Para obtener más información, consulte the section called “Puntos de conexión de VPC”. Sin embargo, para que la canalización escriba en la colección, la política también debe conceder acceso al punto de conexión de VPC que OpenSearch Ingestion crea automáticamente entre la canalización y la colección. Por lo tanto, si elige una colección de

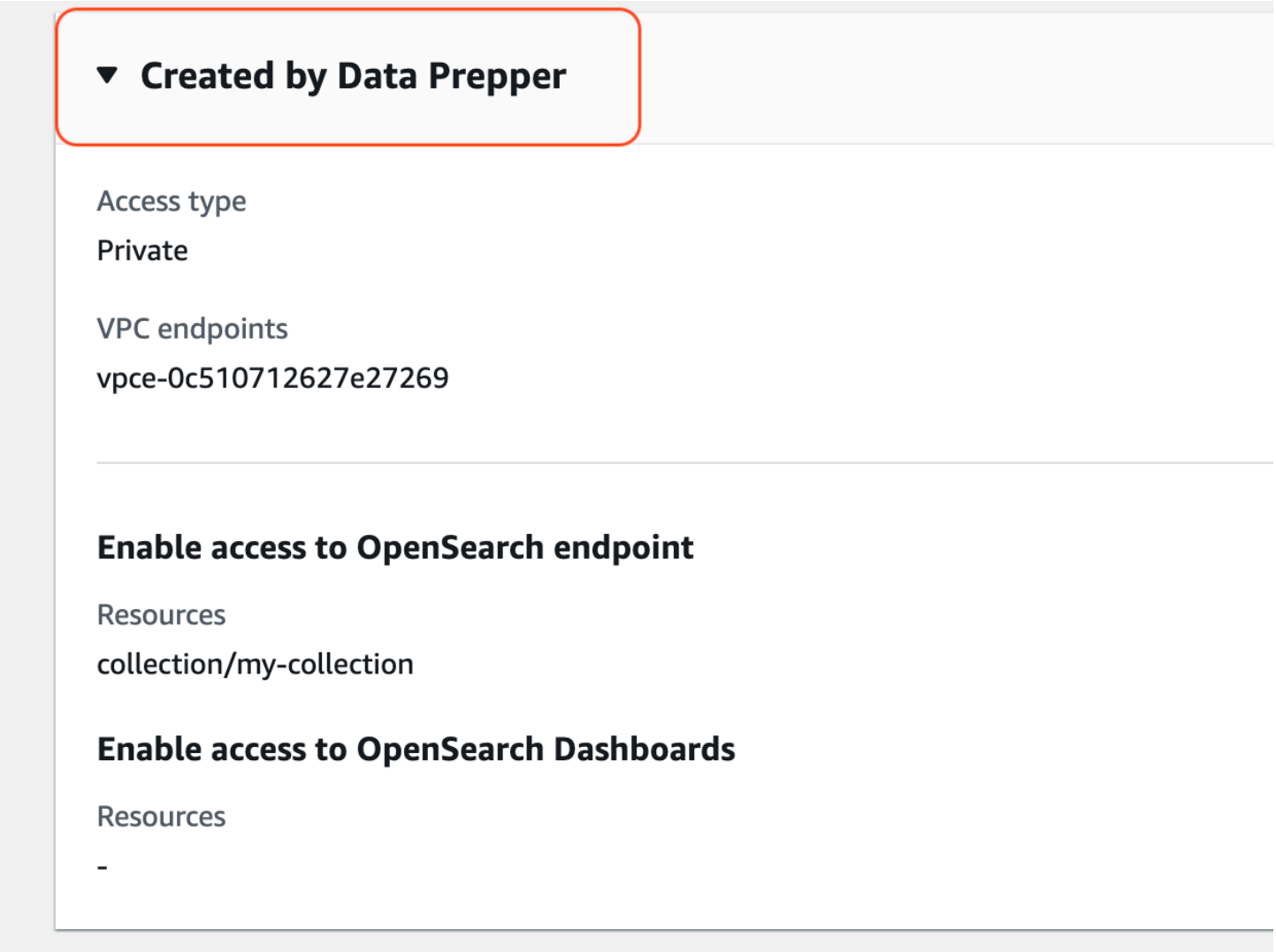
OpenSearch sin servidor como receptor de destino de una canalización, debe escribir el nombre de la política de red asociada en el campo Nombre de la política de red.

Durante la creación de la canalización, OpenSearch Ingestion comprueba la existencia de la política de red especificada. Si no existe, OpenSearch Ingestion la crea. Si existe, OpenSearch Ingestion le agrega una nueva regla para actualizarla. La regla concede acceso al punto de conexión de VPC que conecta la canalización y la colección.

Por ejemplo:

```
{
   "Rules":[
      {
         "Resource":[
            "collection/my-collection"
         ],
         "ResourceType":"collection"
      }
   ],
   "SourceVPCEs":[
      "vpce-0c510712627e27269" # The ID of the VPC endpoint that OpenSearch Ingestion
 creates between the pipeline and collection
   ],
   "Description":"Created by Data Prepper"
}
```

En la consola, cualquier regla que OpenSearch Ingestion agregue a las políticas de red se denomina Creada por Data Prepper:



> **Note**
>
> En general, una regla que especifica el acceso público a una colección anula una regla que especifica el acceso privado. Por lo tanto, si la política ya tenía configurado el acceso público, esta nueva regla que agrega OpenSearch Ingestion, en realidad, no cambia el comportamiento de la política. Para obtener más información, consulte the section called "Prioridad política".

Si detiene o elimina la canalización, OpenSearch Ingestion elimina el punto de conexión de VPC entre la canalización y la colección. También modifica la política de red para eliminar el punto de conexión de VPC de la lista de puntos de conexión permitidos. Si reinicia la canalización, se vuelve

a crear el punto de conexión de VPC y se vuelve a actualizar la política de red con el ID del punto de conexión.

Introducción a Amazon OpenSearch Ingestion

Amazon OpenSearch Ingestion admite la ingesta de datos en dominios gestionados de OpenSearch Service y en colecciones OpenSearch sin servidor. Los tutoriales siguientes muestran los pasos básicos para poner en marcha una canalización.

En el primer tutorial se muestra cómo utilizar Amazon OpenSearch Ingestion para configurar una canalización sencilla e ingerir datos en un dominio de Amazon OpenSearch Service.

En el segundo tutorial se muestra cómo utilizar Amazon OpenSearch Ingestion para configurar una canalización sencilla e ingerir datos en una colección de Amazon OpenSearch sin servidor.

> **Note**
>
> La creación de canalizaciones fallará si no configura los permisos correctos. Consulte the section called “Configuración de roles y usuarios” para comprender mejor las funciones necesarias antes de crear una canalización.

Temas

- Tutorial: incorporar datos a un dominio mediante Amazon OpenSearch Ingestion
- Tutorial: incorporar datos en una colección mediante Amazon OpenSearch Ingestion

Tutorial: incorporar datos a un dominio mediante Amazon OpenSearch Ingestion

En este tutorial se muestra cómo utilizar Amazon OpenSearch Ingestion para configurar una canalización sencilla e incorporar datos en un dominio de Amazon OpenSearch Service. Una canalización es un recurso que OpenSearch Ingestion aprovisiona y administra. Puede usar una canalización para filtrar, enriquecer, transformar, normalizar y agregar datos para el análisis y la visualización posteriores en OpenSearch Service.

Este tutorial recorre los pasos básicos para poner en marcha una canalización de forma rápida. Para obtener instrucciones más completas, consulte the section called “Creación de canalizaciones”.

En este tutorial, deberá completar los siguientes pasos:

1. Crear un dominio.
2. Crear una canalización.
3. Incorporar algunos datos de muestra.

En este tutorial creará los recursos siguientes:

- Un dominio llamado `ingestion-domain` al que la canalización escribe.
- Una canalización llamada `ingestion-pipeline`

Permisos necesarios

Para completar este tutorial, su usuario o rol debe tener una política basada en la identidad adjunta con los siguientes permisos mínimos. Estos permisos le permiten crear un rol de canalización y adjuntar una política (`iam:Create*` y `iam:Attach*`), crear o modificar un dominio (`es:*`) y trabajar con canalizaciones (`osis:*`).

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Resource":"*",
         "Action":[
            "osis:*",
            "iam:Create*",
            "iam:Attach*",
            "es:*"
         ]
      },
      {
         "Resource":[
            "arn:aws:iam::111122223333:role/OpenSearchIngestion-PipelineRole"
         ],
         "Effect":"Allow",
         "Action":[
```

```
            "iam:CreateRole",
            "iam:AttachRolePolicy",
            "iam:PassRole"
        ]
      }
   ]
}
```

Paso 1: crear el rol de canalización

En primer lugar, cree un rol que asuma la canalización para acceder al receptor de dominio de OpenSearch Service. Incluirá esta función en la configuración de la canalización más adelante en este tutorial.

Cómo crear el rol de canalización

1. Abra la consola AWS Identity and Access Management en https://console.aws.amazon.com/iamv2/.
2. Seleccione Políticas y, a continuación, seleccione Crear política.
3. En este tutorial, incorporará datos a un dominio llamado `ingestion-domain`, que creará en el siguiente paso. Seleccione JSON y pegue la siguiente política en el editor. Sustituya *`your-account-id`* por su Identificador de cuenta y modifique la región si es necesario.

   JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "es:DescribeDomain",
            "Resource": "arn:aws:es:us-east-1:111122223333:domain/ingestion-
domain"
        },
        {
            "Effect": "Allow",
            "Action": "es:ESHttp*",
            "Resource": "arn:aws:es:us-east-1:111122223333:domain/ingestion-
domain/*"
        }
```

```
    ]
}
```

Si desea escribir datos en un dominio existente, sustituya `ingestion-domain` por el nombre de su dominio.

> **Note**
>
> Para simplificar este tutorial, utilizaremos una política de acceso amplia. No obstante, en entornos de producción, se recomienda que se aplique una política de acceso más restrictiva a su rol de canalización. Para ver un ejemplo de política que proporciona los permisos mínimos necesarios, consulte the section called “Concesión de acceso a dominios para canalizaciones”.

4. Seleccione Siguiente, seleccione Siguiente y asigne a su política el nombre pipeline-policy.
5. Seleccione Crear política.
6. A continuación, cree un rol y añádale la política. Seleccione Roles y, a continuación, Crear rol.
7. Seleccione Política de confianza personalizada y pegue la siguiente política en el editor:

   JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Principal":{
            "Service":"osis-pipelines.amazonaws.com"
         },
         "Action":"sts:AssumeRole"
      }
   ]
}
```

8. Seleccione Siguiente. A continuación, busque y seleccione pipeline-policy (que acaba de crear).
9. Seleccione Siguiente y asigne al rol el nombre PipelineRole.
10. Seleccione Crear rol.

Recuerde el nombre de recurso de Amazon (ARN) del rol (por ejemplo,). , `arn:aws:iam::your-account-id:role/PipelineRole`). Lo necesitará cuando cree la canalización.

Paso 2: crear un dominio

Primero, cree un dominio denominado `ingestion-domain` al que desee incorporar datos.

Vaya a la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home y cree un dominio que cumpla los siguientes requisitos:

- Funciona en OpenSearch 1.0 o superior, o Elasticsearch 7.4 o superior
- Utiliza el acceso público
- No utiliza el control de acceso detallado

> **Note**
>
> Estos requisitos están pensados para garantizar la simplicidad de este tutorial. En los entornos de producción, puede configurar un dominio con acceso a VPC o utilizar un control de acceso detallado. Para utilizar un control de acceso detallado, consulte Map the pipeline role.

El dominio debe tener una política de acceso que otorgue permiso al rol de IAM `OpenSearchIngestion-PipelineRole`, que OpenSearch Service creará para usted en el siguiente paso. La canalización asumirá este rol para enviar datos al receptor de dominios.

Asegúrese de que el dominio tenga la siguiente política de acceso a nivel de dominio, que otorga al rol de canalización acceso al dominio. Sustituya la Región y el ID de cuenta por los suyos:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/OpenSearchIngestion-PipelineRole"
```

```
      },
      "Action": "es:*",
      "Resource": "arn:aws:es:us-east-1:111122223333:domain/ingestion-domain/*"
    }
  ]
}
```

Para obtener más información sobre cómo crear políticas de acceso a nivel de dominio, consulte the section called “Políticas basadas en recursos”.

Si ya ha creado un dominio, modifique su política de acceso actual para proporcionar los permisos anteriores a `OpenSearchIngestion-PipelineRole`.

Paso 3: crear una canalización

Ahora que tiene un dominio, puede crear una canalización.

Para crear una canalización

1. En la consola de Amazon OpenSearch Service, seleccione Canalizaciones en el panel de navegación izquierdo.
2. Seleccione Create pipeline.
3. Seleccione la canalización en blanco y, a continuación, elija Seleccionar esquema.
4. En este tutorial, crearemos una canalización simple que utiliza el complemento de origen HTTP. El complemento acepta datos de registro en un formato de matriz JSON. Especificaremos un único dominio de OpenSearch Service como destino e incorporaremos todos los datos en el índice `application_logs`.

   En el menú Fuente, seleccione HTTP. Para la ruta, ingrese /logs.
5. Para simplificar este tutorial, configuraremos el acceso público para la canalización. En las opciones de red de origen, seleccione Acceso público. Para obtener información acerca de la configuración del acceso a VPC, consulte the section called “Configuración del acceso a la VPC para canalizaciones”.
6. Elija Siguiente.
7. En Procesador, introduzca la fecha y seleccione Agregar.
8. Habilite Desde la hora de recepción. Deje el resto de opciones con sus valores predeterminados.
9. Elija Siguiente.

10. Configure los detalles del destino. Para el tipo de recurso de OpenSearch, elija Clúster administrado. Luego, elija el dominio de OpenSearch Service que usted creó en la sección anterior.

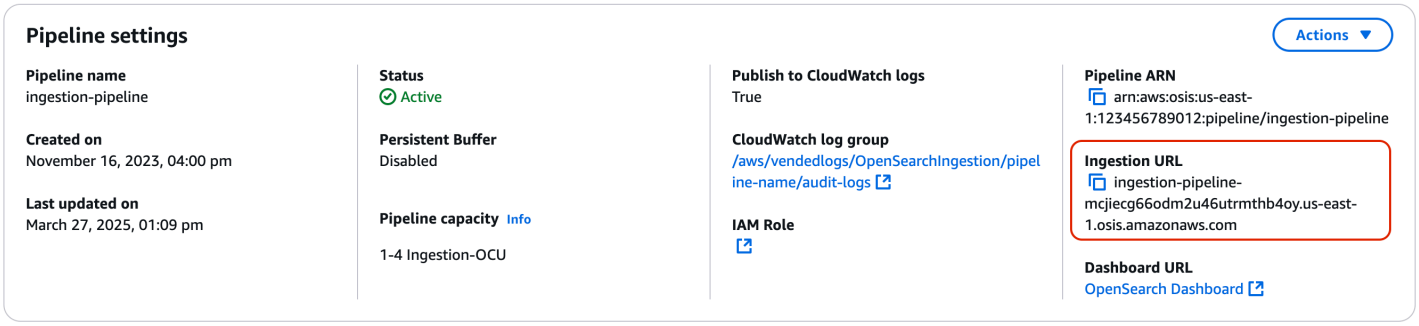
    En Nombre de índice, escriba application_logs. OpenSearch Ingestion crea automáticamente este índice en el dominio si aún no existe.
11. Elija Siguiente.
12. Asigne a la canalización el nombre ingestion-pipeline. Deje la configuración de capacidad con sus valores predeterminados.
13. Para el rol de canalización, seleccione Crear y usar un nuevo rol de servicio. El rol de canalización proporciona los permisos necesarios para que una canalización escriba en el destino del dominio y lea desde orígenes basados en extracción. Al seleccionar esta opción, usted permite que OpenSearch Ingestion cree el rol por usted, en lugar de crearlo manualmente en IAM. Para obtener más información, consulte the section called “Configuración de roles y usuarios”.
14. Para el sufijo del nombre del rol de servicio, ingrese PipelineRole. En IAM, el rol tendrá el formato `arn:aws:iam::your-account-id:role/OpenSearchIngestion-PipelineRole`.
15. Elija Siguiente. Revise la configuración de la canalización y elija Crear canalización. La canalización tarda entre 5 y 10 minutos en activarse.

Paso 4: incorporar algunos datos de muestra

Cuando el estado de la canalización sea `Active`, puede empezar a incorporarle datos. Debe firmar todas las solicitudes HTTP que se envíen a la canalización mediante la versión 4 de Signature. Use una herramienta HTTP como Postman o awscurl para enviar algunos datos a la canalización. Al igual que ocurre con la indexación de datos directamente en un dominio, la incorporación de datos a una canalización siempre requiere un rol de IAM o una clave de acceso y una clave secreta de IAM.

> **Note**
>
> La entidad principal que firma la solicitud debe tener el permiso de IAM `osis:Ingest`.

Primero, obtenga la URL de incorporación en la página de Configuración de canalización:



A continuación, incorpore algunos datos de muestra. La siguiente solicitud utiliza awscurl para enviar un único archivo de registro a la canalización:

```
awscurl --service osis --region us-east-1 \
    -X POST \
    -H "Content-Type: application/json" \
    -d
 '[{"time":"2014-08-11T11:40:13+00:00","remote_addr":"122.226.223.69","status":"404","request":
 http://www.k2proxy.com//hello.html HTTP/1.1","http_user_agent":"Mozilla/4.0
 (compatible; WOW64; SLCC2;)"}]' \
    https://pipeline-endpoint.us-east-1.osis.amazonaws.com/logs
```

Debería ver una respuesta `200 OK`. Si recibe un error de autenticación, es posible que se deba a que está incorporando datos de una cuenta distinta a la de la canalización. Consulte the section called “Solución de problemas de permisos”.

Ahora, consulte el índice `application_logs` para asegurarse de que la entrada de registro se haya incorporado correctamente:

```
awscurl --service es --region us-east-1 \
     -X GET \
     https://search-ingestion-domain.us-east-1.es.amazonaws.com/application_logs/
_search | json_pp
```

Respuesta de ejemplo:

```
{
   "took":984,
   "timed_out":false,
   "_shards":{
      "total":1,
      "successful":5,
```

```
      "skipped":0,
      "failed":0
   },
   "hits":{
      "total":{
         "value":1,
         "relation":"eq"
      },
      "max_score":1.0,
      "hits":[
         {
            "_index":"application_logs",
            "_type":"_doc",
            "_id":"z6VY_IMBRpceX-DU6V4O",
            "_score":1.0,
            "_source":{
               "time":"2014-08-11T11:40:13+00:00",
               "remote_addr":"122.226.223.69",
               "status":"404",
               "request":"GET http://www.k2proxy.com//hello.html HTTP/1.1",
               "http_user_agent":"Mozilla/4.0 (compatible; WOW64; SLCC2;)",
               "@timestamp":"2022-10-21T21:00:25.502Z"
            }
         }
      ]
   }
}
```


Solución de problemas de permisos

Si ha seguido los pasos del tutorial y sigue viendo errores de autenticación cuando intenta incorporar datos, puede que se deba a que la función que escribe en una canalización Cuenta de AWS no es la canalización misma. En este caso, debe crear y asumir un rol que le permita específicamente incorporar datos. Para obtener instrucciones, consulte the section called “Provisión de acceso de ingesta entre cuentas”.

Recursos relacionados

Este tutorial presenta un caso práctico sencillo de incorporación de un único documento a través de HTTP. En escenarios de producción, configurará las aplicaciones de cliente (como Fluent Bit, Kubernetes u OpenTelemetry Collector) para enviar datos a una o más canalizaciones. Es probable que sus canalizaciones sean más complejas que en el ejemplo sencillo de este tutorial.

Para empezar a configurar sus clientes e incorporar datos, consulte los siguientes recursos:

- Creación y administración de canalizaciones
- Configuración de clientes para enviar datos a OpenSearch Ingestion
- Documentación de Data Prepper

Tutorial: incorporar datos en una colección mediante Amazon OpenSearch Ingestion

En este tutorial se muestra cómo utilizar Amazon OpenSearch Ingestion para configurar una canalización sencilla e incorporar datos en una colección de Amazon OpenSearch sin servidor. Una canalización es un recurso que OpenSearch Ingestion aprovisiona y administra. Puede usar una canalización para filtrar, enriquecer, transformar, normalizar y agregar datos para el análisis y la visualización posteriores en OpenSearch Service.

Para ver un tutorial que muestra cómo incorporar datos en un dominio de OpenSearch Service aprovisionado, consulte the section called “Tutorial: incorporar datos a un dominio”.

En este tutorial, deberá completar los siguientes pasos:

1. Crear una colección.
2. Crear una canalización.
3. Incorporar algunos datos de muestra.

En este tutorial creará los recursos siguientes:

- Una colección llamada `ingestion-collection` donde escribirá la canalización
- Una canalización llamada `ingestion-pipeline-serverless`

Permisos necesarios

Para completar este tutorial, su usuario o rol debe tener una política basada en identidad adjunta con los siguientes permisos mínimos. Estos permisos le permiten crear un rol de canalización y adjuntar una política (`iam:Create*` y `iam:Attach*`), crear o modificar una colección (`aoss:*`) y trabajar con canalizaciones (`osis:*`).

Además, se requieren varios permisos de IAM para crear automáticamente el rol de canalización y transferirlo a OpenSearch Ingestion para que pueda escribir datos en la colección.

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Resource":"*",
         "Action":[
            "osis:*",
            "iam:Create*",
            "iam:Attach*",
            "aoss:*"
         ]
      },
      {
         "Resource":[
            "arn:aws:iam::111122223333:role/OpenSearchIngestion-PipelineRole"
         ],
         "Effect":"Allow",
         "Action":[
            "iam:CreateRole",
            "iam:AttachRolePolicy",
            "iam:PassRole"
         ]
      }
   ]
}
```

Paso 1: crear una colección

A continuación, cree una colección para incorporar datos. El nombre que le pondremos a la colección será `ingestion-collection`.

1. Vaya a la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. Seleccione Colecciones del panel de navegación de la izquierda y elija Crear colección.

3. Nombre la colección ingestion-collection.
4. En Seguridad, elija Creación estándar.
5. En Configuración de acceso a la red, cambie el tipo de acceso a Público.
6. Mantenga todas las demás configuraciones en sus valores predeterminados y elija Siguiente.
7. Ahora, configure una política de acceso a los datos para la colección. Anule la selección de Coincidir automáticamente la configuración de la política de acceso.
8. Seleccione JSON para el Método de definición y pegue la siguiente política en el editor. Esta política realiza dos tareas:

   - Permite que el rol de canalización escriba en la colección.
   - Le permite leer de la colección. Más adelante, después de incorporar algunos datos de muestra a la canalización, consultará la colección para garantizar que los datos se hayan incorporado y escrito correctamente en el índice.

```
[
  {
    "Rules": [
      {
        "Resource": [
          "index/ingestion-collection/*"
        ],
        "Permission": [
          "aoss:CreateIndex",
          "aoss:UpdateIndex",
          "aoss:DescribeIndex",
          "aoss:ReadDocument",
          "aoss:WriteDocument"
        ],
        "ResourceType": "index"
      }
    ],
    "Principal": [
      "arn:aws:iam::your-account-id:role/OpenSearchIngestion-PipelineRole",
      "arn:aws:iam::your-account-id:role/Admin"
    ],
    "Description": "Rule 1"
  }
]
```

9. Modifique los elementos `Principal` para incluir su ID de Cuenta de AWS. Para la segunda entidad principal, especifique un usuario o rol que pueda utilizar para consultar la colección más tarde.
10. Elija Siguiente. Nombre la política de acceso pipeline-collection-access y elija Siguiente de nuevo.
11. Revise la configuración de la colección y seleccione Enviar.

Paso 2: crear una canalización

Ahora que tiene una colección, puede crear una canalización.

Para crear una canalización

1. En la consola de Amazon OpenSearch Service, seleccione Canalizaciones en el panel de navegación izquierdo.
2. Seleccione Create pipeline.
3. Seleccione la canalización en blanco y, a continuación, elija Seleccionar esquema.
4. En este tutorial, crearemos una canalización simple que utiliza el complemento de origen HTTP. El complemento acepta datos de registro en un formato de matriz JSON. Especificaremos una única colección de OpenSearch sin servidor como el receptor e incorporaremos todos los datos al índice `my_logs`.

   En el menú Fuente, seleccione HTTP. Para la ruta, ingrese /logs.
5. Para simplificar este tutorial, configuraremos el acceso público para la canalización. En las opciones de red de origen, seleccione Acceso público. Para obtener información acerca de la configuración del acceso a VPC, consulte the section called “Configuración del acceso a la VPC para canalizaciones”.
6. Elija Siguiente.
7. En Procesador, introduzca la fecha y seleccione Agregar.
8. Habilite Desde la hora de recepción. Deje el resto de opciones con sus valores predeterminados.
9. Elija Siguiente.
10. Configure los detalles del destino. Para el tipo de recurso de OpenSearch, elija Colección (sin servidor). Luego, elija la colección de OpenSearch Service que creó en la sección anterior.

Deje el nombre de la política de red como predeterminado. En Nombre de índice, escriba my_logs. OpenSearch Ingestion crea automáticamente este índice en la colección si aún no existe.

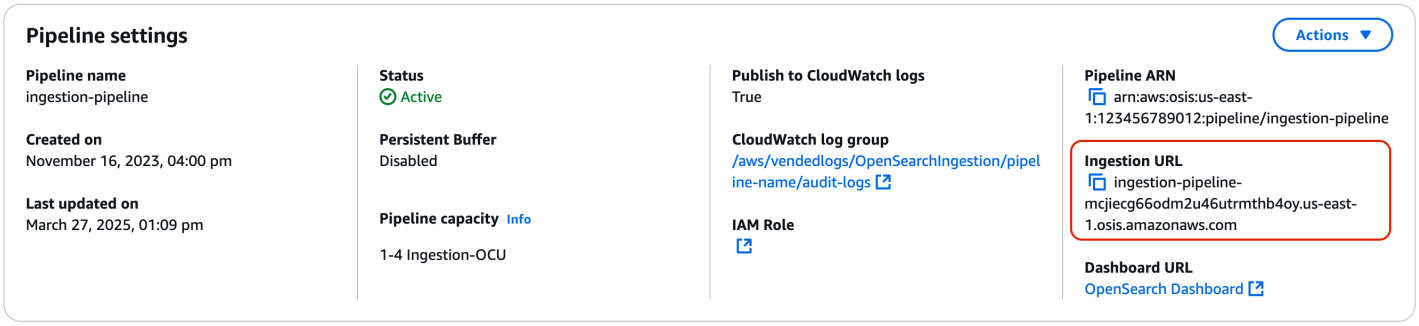
11. Elija Siguiente.
12. Asigne el nombre ingestion-pipeline-serverless a la canalización. Deje la configuración de capacidad con sus valores predeterminados.
13. Para el rol de canalización, seleccione Crear y usar un nuevo rol de servicio. El rol de canalización proporciona los permisos necesarios para que una canalización escriba en el destino de colección y lea desde orígenes basados en extracción. Al seleccionar esta opción, usted permite que OpenSearch Ingestion cree el rol por usted, en lugar de crearlo manualmente en IAM. Para obtener más información, consulte the section called “Configuración de roles y usuarios”.
14. Para el sufijo del nombre del rol de servicio, ingrese PipelineRole. En IAM, el rol tendrá el formato `arn:aws:iam::your-account-id:role/OpenSearchIngestion-PipelineRole`.
15. Elija Siguiente. Revise la configuración de la canalización y elija Crear canalización. La canalización tarda entre 5 y 10 minutos en activarse.

Paso 3: incorporar algunos datos de muestra

Cuando el estado de la canalización sea `Active`, puede empezar a incorporarle datos. Debe firmar todas las solicitudes HTTP que se envíen a la canalización mediante la versión 4 de Signature. Use una herramienta HTTP como Postman o awscurl para enviar algunos datos a la canalización. Al igual que ocurre con la indexación de datos directamente en una colección, la incorporación de datos a una canalización siempre requiere un rol de IAM o una clave de acceso y una clave secreta de IAM.

> **Note**
>
> La entidad principal que firma la solicitud debe tener el permiso de IAM `osis:Ingest`.

Primero, obtenga la URL de incorporación en la página de Configuración de canalización:



A continuación, envíe algunos datos de muestra a la ruta de ingesta. La siguiente solicitud de muestra utiliza awscurl para enviar un único archivo de registro a la canalización:

```
awscurl --service osis --region us-east-1 \
    -X POST \
    -H "Content-Type: application/json" \
    -d
 '[{"time":"2014-08-11T11:40:13+00:00","remote_addr":"122.226.223.69","status":"404","request":
 http://www.k2proxy.com//hello.html HTTP/1.1","http_user_agent":"Mozilla/4.0
 (compatible; WOW64; SLCC2;)"}]' \
    https://pipeline-endpoint.us-east-1.osis.amazonaws.com/logs
```

Debería ver una respuesta `200 OK`.

Ahora, consulte el índice `my_logs` para asegurarse de que la entrada del registro se haya incorporado correctamente:

```
awscurl --service aoss --region us-east-1 \
     -X GET \
     https://collection-id.us-east-1.aoss.amazonaws.com/my_logs/_search | json_pp
```

Respuesta de ejemplo:

```
{
   "took":348,
   "timed_out":false,
   "_shards":{
      "total":0,
      "successful":0,
      "skipped":0,
      "failed":0
```

```
   },
   "hits":{
      "total":{
         "value":1,
         "relation":"eq"
      },
      "max_score":1.0,
      "hits":[
         {
            "_index":"my_logs",
            "_id":"1%3A0%3ARJgDvIcBTy5m12xrKE-y",
            "_score":1.0,
            "_source":{
               "time":"2014-08-11T11:40:13+00:00",
               "remote_addr":"122.226.223.69",
               "status":"404",
               "request":"GET http://www.k2proxy.com//hello.html HTTP/1.1",
               "http_user_agent":"Mozilla/4.0 (compatible; WOW64; SLCC2;)",
               "@timestamp":"2023-04-26T05:22:16.204Z"
            }
         }
      ]
   }
}
```


Recursos relacionados

Este tutorial presenta un caso práctico sencillo de incorporación de un único documento a través de HTTP. En escenarios de producción, configurará las aplicaciones de cliente (como Fluent Bit, Kubernetes u OpenTelemetry Collector) para enviar datos a una o más canalizaciones. Es probable que sus canalizaciones sean más complejas que en el ejemplo sencillo de este tutorial.

Para empezar a configurar sus clientes e incorporar datos, consulte los siguientes recursos:

- Creación y administración de canalizaciones
- Configuración de clientes para enviar datos a OpenSearch Ingestion
- Documentación de Data Prepper

Descripción general de las características de canalización en Amazon OpenSearch Ingestion

Amazon OpenSearch Ingestion aprovisiona canalizaciones, que consisten en un origen, un búfer, cero o más procesadores y uno o más receptores. Las canalizaciones de incorporación funcionan con Data Prepper como motor de datos. Para obtener información general sobre los distintos componentes de una canalización, consulte the section called “Conceptos clave”.

En las secciones siguientes se incluye información general sobre algunas de las características más utilizadas en Amazon OpenSearch Ingestion.

> **Note**
>
> No es una lista exhaustiva de las características disponibles para las canalización. Para obtener una documentación completa de todas las funciones de canalización disponibles, consulte Documentación de Data Prepper. Tenga en cuenta que OpenSearch Ingestion impone algunas restricciones a los complementos y opciones que puede usar. Para obtener más información, consulte the section called “Complementos y opciones compatibles”.

Temas

- Almacenamiento en búfer persistente
- División
- Encadenar
- Colas de mensajes fallidos
- Administración de índices
- Reconocimiento integral
- Contrapresión del origen

Almacenamiento en búfer persistente

Un búfer persistente almacena los datos en un búfer basado en disco en varias zonas de disponibilidad para mejorar la durabilidad de los datos. Puede utilizar el almacenamiento en búfer persistente para ingerir datos de todos los orígenes basados en push compatibles sin configurar un búfer independiente. Estos orígenes incluyen HTTP y OpenTelemetry para los registros, los rastros

y las métricas. Para habilitar el almacenamiento en búfer persistente, seleccione Habilitar búfer persistente al crear o actualizar una canalización. Para obtener más información, consulte the section called “Creación de canalizaciones”.

OpenSearch Ingestion determina de forma dinámica el número de OCU que se van a utilizar para el almacenamiento en búfer persistente, teniendo en cuenta el origen de datos, las transformaciones de transmisión y el destino del receptor. Como asigna algunas OCU al almacenamiento en búfer, es posible que tenga que aumentar las OCU mínimas y máximas para mantener el mismo rendimiento de ingesta. Las canalizaciones conservan los datos en el búfer hasta 72 horas.

Si habilita el almacenamiento en búfer persistente para una canalización, los tamaños máximos de carga útil de solicitud predeterminados son los siguientes:

- Orígenes HTTP: 10 MB
- Orígenes de OpenTelemetry: 4 MB

En el caso de los orígenes HTTP, puede aumentar el tamaño máximo de carga útil a 20 MB. El tamaño de la carga útil de la solicitud incluye toda la solicitud HTTP, que normalmente contiene varios eventos. Cada evento no puede superar los 3,5 MB.

Las canalizaciones con el almacenamiento en búfer persistente dividen las unidades de canalización configuradas en unidades de computación y unidades de búfer. Si una canalización utiliza un procesador con un uso intensivo de la CPU, como grok, key-value o split string, asigna las unidades en una proporción de búfer/proceso de 1:1. De lo contrario, las asigna en una proporción de 3:1, lo que siempre favorece las unidades de cómputo.

Por ejemplo:

- Canalización con grok y 2 unidades como máximo: 1 unidad de computación y 1 unidad de búfer
- Canalización con grok y 5 unidades como máximo: 3 unidades de computación y 2 unidades de búfer
- Canalización sin procesadores y 2 unidades como máximo: 1 unidad de computación y 1 unidad de búfer
- Canalización sin procesadores y 4 unidades como máximo: 1 unidad de computación y 3 unidades de búfer
- Canalización con grok y 5 unidades como máximo: 2 unidades de computación y 3 unidades de búfer

De forma predeterminada, las canalizaciones utilizan una Clave propiedad de AWS para cifrar los datos del búfer. Estas canalizaciones no necesitan ningún permiso adicional para el rol de canalización.

Como alternativa, se puede especificar una clave administrada por el cliente y agregar los siguientes permisos de IAM al rol de canalización:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "KeyAccess",
            "Effect": "Allow",
            "Action": [
              "kms:Decrypt",
              "kms:GenerateDataKeyWithoutPlaintext"
            ],
            "Resource": "arn:aws:kms:us-
east-1:111122223333:key/ASIAIOSFODNN7EXAMPLE"
        }
    ]
}
```

Para más información, consulte las claves administradas por el cliente en la Guía para desarrolladores de AWS Key Management Service.

> **Note**
>
> Si deshabilita el almacenamiento en búfer persistente, su canalización se ejecuta por completo con el almacenamiento en búfer en memoria.

División

Puede configurar una canalización de OpenSearch Ingestion para dividir los eventos entrantes en una subcanalización, lo que le permitirá realizar diferentes tipos de procesamiento en el mismo evento entrante.

El siguiente ejemplo de canalización divide los eventos entrantes en dos subcanalizaciones. Cada subcanalización utiliza su propio procesador para enriquecer y manipular los datos y, a continuación, los envía a distintos índices de OpenSearch.

```
version: "2"
log-pipeline:
  source:
    http:
    ...
  sink:
    - pipeline:
        name: "logs_enriched_one_pipeline"
    - pipeline:
        name: "logs_enriched_two_pipeline"

logs_enriched_one_pipeline:
  source:
    log-pipeline
  processor:
   ...
  sink:
    - opensearch:
        # Provide a domain or collection endpoint
        # Enable the 'serverless' flag if the sink is an OpenSearch Serverless
 collection
        aws:
          ...
        index: "enriched_one_logs"

logs_enriched_two_pipeline:
  source:
    log-pipeline
  processor:
   ...
  sink:
    - opensearch:
        # Provide a domain or collection endpoint
        # Enable the 'serverless' flag if the sink is an OpenSearch Serverless
 collection
        aws:
          ...
          index: "enriched_two_logs"
```

Encadenar

Puede encadenar varias subcanalizaciones para procesar y enriquecer los datos por partes. En otras palabras, puede enriquecer un evento entrante con ciertas capacidades de procesamiento en una subcanalización, luego enviarlo a otra subcanalización para un enriquecimiento adicional con un procesador diferente, y finalmente enviarlo a su receptor de OpenSearch.

En el siguiente ejemplo, la subcanalización `log_pipeline` enriquece un evento de registro entrante con un conjunto de procesadores y, a continuación, envía el evento a un índice de OpenSearch denominado `enriched_logs`. La canalización envía el mismo evento a la subcanalización de `log_advanced_pipeline`, que lo procesa y lo envía a un índice de OpenSearch diferente denominado `enriched_advanced_logs`.

```
version: "2"
log-pipeline:
  source:
    http:
    ...
  processor:
    ...
  sink:
    - opensearch:
        # Provide a domain or collection endpoint
        # Enable the 'serverless' flag if the sink is an OpenSearch Serverless
 collection
        aws:
          ...
          index: "enriched_logs"
    - pipeline:
        name: "log_advanced_pipeline"

log_advanced_pipeline:
  source:
    log-pipeline
  processor:
   ...
  sink:
    - opensearch:
        # Provide a domain or collection endpoint
        # Enable the 'serverless' flag if the sink is an OpenSearch Serverless
 collection
        aws:
```

```
        ...
        index: "enriched_advanced_logs"
```

Colas de mensajes fallidos

Las colas de mensajes fallidos (DLQ) son los destinos de los eventos que una canalización no logra registrar en un receptor. En OpenSearch Ingestion, debe especificar un bucket de Amazon S3 con los permisos de escritura adecuados para usarlo como DLQ. Puede añadir una configuración de DLQ a cada receptor de una canalización. Cuando una canalización detecta errores de escritura, crea objetos de DLQ en el bucket de S3 configurado. Los objetos DLQ existen dentro de un archivo JSON como una matriz de eventos fallidos.

Una canalización escribe eventos en la DLQ cuando se cumple alguna de las siguientes condiciones:

- Se ha agotado el número máximo de reintentos del receptor de OpenSearch. OpenSearch Ingestion requiere un mínimo de 16 para esta configuración.
- El receptor rechaza los eventos debido a una condición de error.

Configuración

Para configurar una cola de mensajes fallidos para una subcanalización, elija Habilitar DLQ de S3 al configurar el destino de su receptor. A continuación, especifique los ajustes necesarios para la cola. Para obtener más información, consulte Configuración en la documentación de Data Prepper DLQ.

Los archivos que se escriben en este DLQ de S3 tienen el siguiente patrón de nomenclatura:

```
dlq-v${version}-${pipelineName}-${pluginId}-${timestampIso8601}-${uniqueId}
```

Para obtener instrucciones acerca de cómo configurar manualmente el rol de canalización para permitir el acceso al bucket de S3 en el que escribe la DLQ, consulte the section called “Permisos para escribir en Amazon S3 o en una cola de mensajes fallidos”.

Ejemplo

Considere el siguiente ejemplo de archivo DLQ:

```
dlq-v2-apache-log-pipeline-opensearch-2023-04-05T15:26:19.152938Z-e7eb675a-
f558-4048-8566-dac15a4f8343
```

Este es un ejemplo de datos que no se pudieron escribir en el receptor y que se envían al bucket DLQ de S3 para su posterior análisis:

```
Record_0
pluginId            "opensearch"
pluginName          "opensearch"
pipelineName        "apache-log-pipeline"
failedData
index    "logs"
indexId   null
status    0
message  "Number of retries reached the limit of max retries (configured value 15)"
document
log      "sample log"
timestamp     "2023-04-14T10:36:01.070Z"

Record_1
pluginId            "opensearch"
pluginName          "opensearch"
pipelineName        "apache-log-pipeline"
failedData
index               "logs"
indexId   null
status    0
message  "Number of retries reached the limit of max retries (configured value 15)"
document
log                 "another sample log"
timestamp           "2023-04-14T10:36:01.071Z"
```

Administración de índices

Amazon OpenSearch Ingestion cuenta con numerosas funciones de administración de índices, entre las que se incluyen las siguientes.

Creación de índices

Puede especificar un nombre de índice en un receptor de canalización y OpenSearch Ingestion crea el índice cuando aprovisiona la canalización. Si ya existe un índice, la canalización lo usa para indexar los eventos entrantes. Si detiene y reinicia una canalización, o si actualiza su configuración de YAML, la canalización intentará crear nuevos índices si aún no existen. Una canalización nunca puede eliminar un índice.

En el siguiente ejemplo, los receptores crean dos índices cuando se aprovisiona la canalización:

```
sink:
  - opensearch:
      index: apache_logs
  - opensearch:
      index: nginx_logs
```

Creación de nombres y patrones de índice

Puede generar nombres de índice dinámicos mediante variables de los campos de los eventos entrantes. En la configuración del receptor, utilice el formato `string${}` para indicar la interpolación de cadenas y utilice un puntero JSON para extraer los campos de los eventos. Las opciones para `index_type` son `custom` o `management_disabled`. Como `index_type` se revierte a `custom` para los dominios de OpenSearch y `management_disabled` para las colecciones de OpenSearch sin servidor, se puede dejar sin configurar.

Por ejemplo, la siguiente canalización selecciona el campo `metadataType` entre los eventos entrantes para generar nombres de índice.

```
pipeline:
  ...
  sink:
    opensearch:
      index: "metadata-${metadataType}"
```

La siguiente configuración sigue generando un índice nuevo cada día o cada hora.

```
pipeline:
  ...
  sink:
    opensearch:
      index: "metadata-${metadataType}-%{yyyy.MM.dd}"

pipeline:
  ...
  sink:
    opensearch:
      index: "metadata-${metadataType}-%{yyyy.MM.dd.HH}"
```

El nombre del índice también puede ser una cadena simple con un patrón de fecha y hora como sufijo, por ejemplo `my-index-%{yyyy.MM.dd}`. Cuando el receptor envía datos a OpenSearch, reemplaza el patrón de fecha y hora por el horario UTC y crea un índice nuevo para cada día, por ejemplo `my-index-2022.01.25`. Para obtener más información, consulte la clase DateTimeFormatter.

Este nombre de índice también puede ser una cadena formateada (con o sin un sufijo de patrón de fecha y hora), como `my-${index}-name`. Cuando el receptor envía datos a OpenSearch, reemplaza la parte `"${index}"` por el valor en caso de que se esté procesando. Si el formato lo es `"${index1/index2/index3}"`, reemplaza el campo `index1/index2/index3` por su valor en el evento.

Creación de identificadores de documentos

Una canalización puede generar un identificador de documento al indexar los documentos en OpenSearch. Puede deducir estos identificadores de documentos a partir de los campos incluidos en los eventos entrantes.

En este ejemplo, se utiliza el campo `uuid` de un evento entrante para generar un identificador de documento.

```
pipeline:
  ...
  sink:
    opensearch:
      index_type: custom
      index: "metadata-${metadataType}-%{yyyy.MM.dd}"
      "document_id": "uuid"
```

En el ejemplo siguiente, el procesador Add entries combina los campos `uuid` y `other_field` del evento entrante para generar un identificador de documento.

La acción `create` garantiza que los documentos con identificadores idénticos no se sobrescriban. La canalización elimina los documentos duplicados sin ningún tipo de reintento ni ningún evento de DLQ. Esta es una expectativa razonable para los autores de canalización que utilizan esta acción, ya que el objetivo es evitar actualizar los documentos existentes.

```
pipeline:
  ...
```

```
  processor:
   - add_entries:
      entries:
        - key: "my_doc_id_field"
          format: "${uuid}-${other_field}"
  sink:
    - opensearch:
       ...
       action: "create"
       document_id: "my_doc_id"
```

Es posible que desee establecer el identificador del documento de un evento en un campo de un subobjeto. En el siguiente ejemplo, el complemento receptor de OpenSearch utiliza el subobjeto `info/id` para generar un identificador de documento.

```
sink:
  - opensearch:
       ...
       document_id: info/id
```

Si se produce el siguiente evento, la canalización generará un documento con el campo `_id` establecido en `json001`:

```
{
   "fieldA":"arbitrary value",
   "info":{
      "id":"json001",
      "fieldA":"xyz",
      "fieldB":"def"
   }
}
```

Creación de identificadores de enrutamiento

Puede usar la opción `routing_field` del complemento del receptor de OpenSearch para establecer el valor de una propiedad de enrutamiento de documentos (`_routing`) en el valor de un evento entrante.

El enrutamiento admite la sintaxis de punteros JSON, por lo que también están disponibles campos anidados, no solo campos de nivel superior.

```
sink:
  - opensearch:
      ...
      routing_field: metadata/id
      document_id: id
```

Si se produce el siguiente evento, el complemento genera un documento con el campo `_routing` establecido en `abcd`:

```
{
   "id":"123",
   "metadata":{
      "id":"abcd",
      "fieldA":"valueA"
   },
   "fieldB":"valueB"
}
```

Para obtener instrucciones sobre cómo crear plantillas de índice que las canalizaciones puedan utilizar durante la creación del índice, consulte Plantillas de índice.

Reconocimiento integral

OpenSearch Ingestion garantiza la durabilidad y la fiabilidad de los datos mediante el seguimiento de su entrega desde el origen hasta los receptores en las canalizaciones sin estado mediante un reconocimiento integral.

> **Note**
>
> Actualmente, solo el complemento fuente de S3 admite el reconocimiento integral.

Con el reconocimiento integral, el complemento de origen de la canalización crea un conjunto de reconocimiento para monitorear un lote de eventos. Recibe un reconocimiento positivo cuando esos eventos se envían correctamente a sus receptores, o un reconocimiento negativo cuando alguno de los eventos no se ha podido enviar a sus receptores.

En caso de fallo o caída de un componente de la canalización, o si un origen no recibe un reconocimiento, se agota el tiempo de espera y toma las medidas necesarias, como volver

a intentarlo o registrar el fallo. Si la canalización tiene configurados varios receptores o subcanalizaciones, las confirmaciones a nivel de evento solo se envían después de que el evento se envíe a todos los receptores de todas las subcanalizaciones. Si un receptor tiene una DLQ configurada, los reconocimientos integrales también rastrean los eventos escritos en la DLQ.

Para habilitar el reconocimiento de extremo a extremo, expanda Opciones adicionales en la configuración de origen de Amazon S3 y elija Habilitar el reconocimiento de mensajes de extremo a extremo.

Contrapresión del origen

Una canalización puede sufrir una contrapresión cuando está ocupada procesando datos o si sus receptores están temporalmente inactivos o tardan en incorporar datos. OpenSearch Ingestion tiene diferentes formas de gestionar la contrapresión en función del complemento origen que utilice la canalización.

Origen de HTTP

Las canalizaciones que utilizan el complemento origen de HTTP gestionan la contrapresión de forma diferente en función del componente de la canalización que esté congestionado:

- Búferes: cuando los búferes están llenos, la canalización comienza a devolver el estado HTTP `REQUEST_TIMEOUT` con el código de error 408 al punto de conexión de origen. A medida que se van liberando los búferes, la canalización vuelve a procesar los eventos HTTP.
- Subprocesos de origen: cuando todos los subprocesos de origen HTTP están ocupados ejecutando solicitudes y el tamaño de la cola de solicitudes sin procesar ha superado el número máximo de solicitudes permitido, la canalización comienza a devolver el estado HTTP `TOO_MANY_REQUESTS` con el código de error 429 al punto de conexión de origen. Cuando la cola de solicitudes cae por debajo del tamaño máximo permitido, la canalización vuelve a procesar las solicitudes.

Origen de OTel

Cuando los búferes están llenos para las canalizaciones que utilizan orígenes de OpenTelemetry (registros de OTel, métricas de OTel y trazas de OTel), la canalización comienza a devolver el estado HTTP `REQUEST_TIMEOUT` con el código de error 408 al punto de conexión de origen. A medida que se van liberando los búferes, la canalización vuelve a procesar los eventos.

Fuente de S3

Cuando los búferes están llenos para las canalizaciones con un origen de S3, las canalizaciones dejan de procesar las notificaciones de SQS. A medida que se van liberando los búferes, la canalización vuelve a procesar las notificaciones.

Si un receptor está inactivo o no puede incorporar datos y está activado el reconocimiento integral para el origen, la canalización deja de procesar las notificaciones de SQS hasta que reciba un reconocimiento correcto de todos los receptores.

Creación de canalizaciones de Amazon OpenSearch Ingestion

Una canalización es el mecanismo que Amazon OpenSearch Ingestion utiliza para mover los datos desde su origen (de donde provienen los datos) hasta su receptor (donde van los datos). En OpenSearch Ingestion, el receptor siempre será un único dominio de Amazon OpenSearch Service, mientras que la fuente de los datos podrían ser clientes como Amazon S3, Fluent Bit o el Recolector de OpenTelemetry.

Para más información, consulte Canalizaciones en la documentación de OpenSearch.

Temas

- Requisitos previos y rol de IAM requerido
- Permisos de IAM necesarios
- Especificar la versión de la canalización
- Especificación de la ruta de ingesta
- Creación de canalizaciones
- Seguimiento del estado de creación de la canalización
- Uso de esquemas

Requisitos previos y rol de IAM requerido

Para crear una canalización de OpenSearch Ingestion, debe disponer de los siguientes recursos:

- Un rol de IAM, llamado rol de canalización, que OpenSearch Ingestion asumirá para poder escribir en el receptor. Puede crear este rol con antelación o puede hacer que OpenSearch Ingestion lo cree automáticamente mientras crea la canalización.

- Un dominio de OpenSearch Service o una colección de OpenSearch sin servidor que actúe como receptor. Si escribe a un dominio, debe ejecutar OpenSearch 1.0 o posterior, Elasticsearch 7.4 o posterior. El receptor debe tener una política de acceso que conceda los permisos adecuados a su rol de canalización de IAM.

Si desea obtener instrucciones para crear estos recursos, consulte los siguientes temas:

- the section called “Concesión de acceso a dominios para canalizaciones”
- the section called “Concesión de acceso a colecciones para canalizaciones”

> **Note**
>
> Si escribe en un dominio que usa un control de acceso detallado, debe completar algunos pasos adicionales. Consulte the section called “Asignar el rol de canalización (solo para dominios que utilicen un control de acceso detallado)”.

Permisos de IAM necesarios

OpenSearch Ingestion utiliza los siguientes permisos de IAM para crear canalizaciones:

- `osis:CreatePipeline`: Crear una canalización.
- `osis:ValidatePipeline`: Comprobar si la configuración de la canalización es válida.
- `iam:CreateRole` y `iam:AttachPolicy`: haga que OpenSearch Ingestion cree automáticamente el rol de canalización por usted.
- `iam:PassRole`: transfiera el rol de canalización a OpenSearch Ingestion para que pueda escribir datos en el dominio. Este permiso debe estar en el recurso de rol de canalización, o simplemente coloque `*` si planea usar diferentes roles en cada canalización.

Por ejemplo, la siguiente política concede permiso para crear una canalización:

JSON

```
{
   "Version":"2012-10-17",
```

```
   "Statement":[
      {
         "Effect":"Allow",
         "Resource":"*",
         "Action":[
            "osis:CreatePipeline",
            "osis:ListPipelineBlueprints",
            "osis:ValidatePipeline"
         ]
      },
      {
         "Resource":[
            "arn:aws:iam::111122223333:role/pipeline-role"
         ],
         "Effect":"Allow",
         "Action":[
            "iam:CreateRole",
            "iam:AttachRolePolicy",
            "iam:PassRole"
         ]
      }
   ]
}
```

OpenSearch Ingestion también incluye un permiso denominado `osis:Ingest`, que es necesario para enviar solicitudes firmadas a la canalización mediante Signature Version 4. Para más información, consulte the section called “Creación de un rol de ingesta”.

> **Note**
>
> Además, el primer usuario que cree una canalización en una cuenta debe tener permisos para realizar la acción `iam:CreateServiceLinkedRole`. Para más información, consulte recurso de rol de canalización.

Para obtener más información sobre cada permiso, consulte Acciones, recursos y claves de condición de OpenSearch Ingestion en la Referencia de autorizaciones de servicio.

Especificar la versión de la canalización

Al crear una canalización con el editor de configuración, debe especificar la versión principal de Data Prepper en la que se ejecutará la canalización. Para especificar la versión, incluya la opción de `version` en la configuración de la canalización:

```
version: "2"
log-pipeline:
  source:
    ...
```

Al elegir Crear, OpenSearch Ingestion determina la última versión secundaria disponible de la versión principal que especifique y aprovisiona la canalización con esa versión. Por ejemplo, si especifica `version: "2"` y la última versión compatible de Data Prepper es la 2.1.1, OpenSearch Ingestion aprovisiona la canalización con la versión 2.1.1. No mostramos públicamente la versión secundaria que está ejecutando la canalización.

Para actualizar la canalización cuando haya disponible una nueva versión principal de Data Prepper, edite la configuración de la canalización y especifique la nueva versión. No puede cambiar una canalización a una versión anterior.

> **Note**
>
> OpenSearch Ingestion no admite inmediatamente las nuevas versiones de Data Prepper tan pronto como se lanzan. Habrá cierto retraso entre el momento en que una nueva versión esté disponible públicamente y el momento en que OpenSearch Ingestion la admita. Además, es posible que OpenSearch Ingestion no sea explícitamente compatible con determinadas versiones principales o secundarias. Para obtener una lista completa, consulte the section called “Versiones de Data Prepper admitidas”.

Cada vez que realice un cambio en su canalización que inicie una implementación azul/verde, OpenSearch Ingestion puede actualizarla a la última versión secundaria de la versión principal que esté configurada actualmente en la canalización. Para obtener más información, consulte the section called “Implementaciones azul/verde para actualizaciones de canalización”. OpenSearch Ingestion no puede cambiar la versión principal de la canalización a menos que actualice explícitamente la opción de `version` en la configuración de la canalización.

Especificación de la ruta de ingesta

Para las fuentes basadas en pull, como el rastreo de OTel y las métricas de OTel, OpenSearch Ingestion requiere la opción de `path` adicional en la configuración de la fuente. La ruta es una cadena, por ejemplo`/log/ingest`, que representa la ruta del URI para la ingesta. Esta ruta define el URI que usa para enviar datos a la canalización.

Por ejemplo, supongamos que especifica la siguiente ruta para una canalización con un origen HTTP:



Al introducir datos de ingesta en la canalización, debe especificar el siguiente punto de conexión en la configuración de su cliente: `https://`*`pipeline-name-abc123`*`.`*`us-west-2`*`.osis.amazonaws.com/`*`my`*`/`*`test_path`*.

La ruta debe empezar con una barra (/) y puede contener los caracteres especiales '-', '_', '.' y '/', así como el marcador de posición `${pipelineName}`. Si utiliza `${pipelineName}` (por ejemplo `/${pipelineName}/test_path`), OpenSearch Ingestion reemplaza la variable con el nombre de la subcanalización asociada.

Creación de canalizaciones

Esta sección describe cómo crear canalizaciones de OpenSearch Ingestion mediante la consola de OpenSearch Service y la AWS CLI.

Consola

Para crear una canalización, inicie sesión en la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home y seleccione Crear canalización.

Seleccione una canalización en blanco o elija un esquema de configuración. Los esquemas incluyen una canalización preconfigurada para una variedad de casos de uso comunes. Para obtener más información, consulte the section called “Uso de esquemas”.

Elija Seleccionar esquema.

Configurar el origen

1. Si parte de una canalización vacía, seleccione un origen del menú desplegable. Los orígenes disponibles pueden incluir otros Servicios de AWS, OpenTelemetry o HTTP. Para obtener más información, consulte the section called “Integración de canalizaciones”.
2. En función del origen que elija, configure los ajustes adicionales correspondientes. Por ejemplo, para utilizar Amazon S3 como origen, debe especificar la URL de la cola de Amazon SQS de la canalización que recibe los mensajes. Para obtener una lista de complementos de origen compatibles y enlaces a su documentación, consulte the section called “Complementos y opciones compatibles”.
3. Para algunos orígenes, debe especificar Opciones de red de origen. Seleccione Acceso a VPC o Acceso público. Si elige Acceso público, vaya al siguiente paso. Si elige Acceso a la VCP, configure los siguientes ajustes:

Opción	Descripción
Administración de puntos de conexión	Elija si quiere crear los puntos de conexión en la nube privada virtual (VPC) o dejar que OpenSearch Ingestion los cree. La administración de puntos de conexión utiliza de forma predeterminada los puntos de conexión administrados por OpenSearch Ingestion.
de VPC	Elija el ID de la VPC que desea utilizar. La VPC y la canalización deben estar en la misma Región de AWS.
Subredes	Seleccione una o varias subredes. OpenSearch Service colocará un punto de conexión de VPC e interfaces de red elásticas en las subredes.
Grupos de seguridad	Seleccione uno o más grupos de seguridad de VPC que permitan que la aplicación requerida llegue a la canalización de OpenSearch Ingestion en los puertos (80 o 443) y protocolos (HTTP o HTTPS) que expone la canalización.
Opciones de conexión de VPC	Si el origen es un punto de conexión autoadministrado, conecte la canalización a una VPC. Elija una de las opciones de CIDR predeterm inadas que se proporcionan o utilice un CIDR personalizado.

Para obtener más información, consulte the section called “Configuración del acceso a la VPC para canalizaciones”.

4. Elija Siguiente.

Configurar el procesador

Agregue uno o más procesadores a su canalización. Los procesadores son componentes de una subcanalización que permiten filtrar, transformar y enriquecer los eventos antes de publicar los registros en el dominio o en el receptor de la colección. Para obtener una lista de los procesadores compatibles y los enlaces a su documentación, consulte the section called “Complementos y opciones compatibles”.

Puede elegir Acciones y agregar lo siguiente:

- Enrutamiento condicional: enruta los eventos a diferentes receptores en función de condiciones específicas. Para obtener más información, consulte Enrutamiento condicional.
- Subcanalización : cada subcanalización es una combinación de un solo origen, cero o más procesadores y un único receptor. Solo una subcanalización puede tener un origen externo. Todas las demás deben tener orígenes que sean otras subcanalizaciones dentro de la configuración general de la canalización. Una única configuración de canalización puede contener de 1 a 10 subcanalizaciones.

Elija Siguiente.

Configurar el receptor

Seleccione el destino en el que la canalización publica los registros. Cada subcanalización debe contener al menos un receptor. Puede agregar un máximo de 10 receptores a una canalización.

Para los receptores de OpenSearch, configure los siguientes campos:

Opción	Descripción
Nombre de la política de red	Si seleccionó una colección OpenSearch sin servidor, escriba un Nombre de política de red. OpenSearch Ingestion crea la política si no existe o la actualiza con una regla que otorga acceso al punto de conexión de VPC de

Opción	Descripción
(Solo receptores sin servidor)	la canalización y la colección. Para obtener más información, consulte the section called “Concesión de acceso a colecciones para canalizaciones”.
Nombre de índice	El nombre del índice al que la canalización envía los datos. OpenSearch Ingestion crea este índice si aún no existe.
Opciones de asignación de índices	Elija cómo la canalización almacena e indexa los documentos y sus campos en el receptor de OpenSearch. Si selecciona Asignación dinámica, OpenSearch agrega campos automáticamente al indexar un documento. Si selecciona Personalizar asignación, ingrese una plantilla de asignación de índices. Para más información, consulte Plantillas de índices.
Habilite DLQ	Configure una cola de mensajes fallidos (DLQ) de Amazon S3 para la canalización. Para obtener más información, consulte the section called “Colas de mensajes fallidos”.
Ajustes adicionales	Configure las opciones avanzadas del receptor de OpenSearch. Para obtener más información, consulte Opciones de configuración en la documentación de Data Prepper.

Para agregar un receptor de Amazon S3, seleccione Agregar receptor y Amazon S3. Para obtener más información, consulte the section called “Amazon S3 como destino”.

Elija Siguiente.

Configuración de la canalización

Configure los siguientes ajustes de canalización adicional:

Opción	Descripción
Nombre de canalización	Un nombre único para la canalización.
Búfer persistente	Un búfer persistente almacena los datos en un búfer basado en disco en varias zonas de disponibilidad. Para obtener más información, consulte the section called “Almacenamiento en búfer persistente”.

Opción	Descripción
	Si habilita el búfer persistente, seleccione la clave AWS Key Management Service para cifrar los datos del búfer.
Capacidad de las canalizaciones	La capacidad mínima y máxima de canalización en las unidades de cómputo (OCU) de Ingestion OpenSearch. Para obtener más información, consulte the section called “Escala de canalizaciones”.
Rol de canalizac ión	El rol de IAM que proporciona los permisos necesarios para que la canalizac ión escriba en el receptor y lea desde orígenes basados en extracciones. Puede crear el rol usted mismo o dejar que OpenSearch Ingestion lo cree por usted en función del caso de uso seleccionado. Para obtener más información, consulte the section called “Configuración de roles y usuarios”.
Etiquetas de	Agregue una o más etiquetas a su canalización. Para obtener más informaci ón, consulte the section called “Etiquetado de canalizaciones”.
Opciones de publicación de registros	Habilite la publicación de registros de canalizaciones en Registros de Amazon CloudWatch. Le recomendamos que habilite la publicación de registros para poder solucionar más fácilmente los problemas de la canalizac ión. Para obtener más información, consulte the section called “Monitoreo de registros de canalización”.

Elija Siguiente, revise la configuración de la canalización y elija Crear canalización.

OpenSearch Ingestion ejecuta un proceso asíncrono para crear la canalización. Una vez que el estado de la canalización sea `Active`, puede empezar a incorporar datos.

AWS CLI

El comando create-pipeline acepta la configuración de la canalización como una cadena o dentro de un archivo .yaml o .json. Si proporciona la configuración en forma de cadena, cada nueva línea debe ir acompañada de secuencias de escape \n. Por ejemplo: ., `"log-pipeline:\n source: \n http:\n processor:\n - grok:\n ...`

El siguiente comando de ejemplo crea una canalización con la siguiente configuración:

- Mínimo de 4 OCU de ingesta, máximo de 10 OCU de ingesta
- Aprovisionada en una nube privada virtual (VPC)
- Publicación de registros habilitada

```
aws osis create-pipeline \
  --pipeline-name my-pipeline \
  --min-units 4 \
  --max-units 10 \
  --log-publishing-options
 IsLoggingEnabled=true,CloudWatchLogDestination={LogGroup="MyLogGroup"} \
  --vpc-options
 SecurityGroupIds={sg-12345678,sg-9012345},SubnetIds=subnet-1212234567834asdf \
  --pipeline-configuration-body "file://pipeline-config.yaml" \
  --pipeline-role-arn  arn:aws:iam::1234456789012:role/pipeline-role
```

OpenSearch Ingestion ejecuta un proceso asíncrono para crear la canalización. Una vez que el estado de la canalización sea `Active`, puede empezar a incorporar datos. Para comprobar el estado de la canalización, use el comando GetPipeline.

API de OpenSearch Ingestion

Para crear una canalización de OpenSearch Ingestion mediante la API de OpenSearch Ingestion, llame a la operación CreatePipeline.

Una vez que la canalización se haya creado correctamente, puede configurar su cliente y empezar a incorporar datos en su dominio de OpenSearch Service. Para más información, consulte the section called “Integración de canalizaciones”.

Seguimiento del estado de creación de la canalización

Puede realizar un seguimiento del estado de una canalización a medida que OpenSearch Ingestion la aprovisiona y la prepara para incorporar datos.

Consola

Después de crear inicialmente una canalización, esta pasa por varias etapas a medida que OpenSearch Ingestion la prepara para la ingesta de datos. Para ver las distintas etapas de creación de la canalización, seleccione el nombre de la canalización para ver la página de Configuración de la canalización. En Estado, seleccione Ver detalles.

Una canalización pasa por las siguientes etapas antes de estar disponible para incorporar datos:

- Validación: se valida la configuración de la canalización. Cuando se complete esta etapa, todas las validaciones se han realizado correctamente.
- Crear entorno: preparar y aprovisionar recursos. Cuando se complete esta etapa, se habrá creado el nuevo entorno de canalización.
- Implementar canalización: implementar la canalización. Cuando se complete esta etapa, la canalización se habrá implementado correctamente.
- Comprobar el estado de la canalización: comprobación del estado de la canalización. Cuando se complete esta etapa, todas las comprobaciones de estado se habrán aprobado.
- Habilitar tráfico: permitir que la canalización incorpore datos. Cuando se complete esta etapa, puede empezar a incorporar datos a la canalización.

CLI

Use el comando get-pipeline-change-progress para comprobar el estado de una canalización. La siguiente solicitud de AWS CLI comprueba el estado de una canalización denominada `my-pipeline`:

```
aws osis get-pipeline-change-progress \
    --pipeline-name my-pipeline
```

Respuesta:

```
{
   "ChangeProgressStatuses": {
      "ChangeProgressStages": [
         {
            "Description": "Validating pipeline configuration",
            "LastUpdated": 1.671055851E9,
            "Name": "VALIDATION",
            "Status": "PENDING"
         }
      ],
      "StartTime": 1.671055851E9,
      "Status": "PROCESSING",
      "TotalNumberOfStages": 5
   }
}
```

API de OpenSearch Ingestion

Para realizar un seguimiento del estado de creación de la canalización mediante la API de OpenSearch Ingestion, llame a la operación GetPipelineChangeProgress.

Uso de esquemas

En lugar de crear una definición de canalización desde cero, puede usar esquemas de configuración, que son plantillas preconfiguradas para escenarios de ingesta comunes, como los registros de Trace Analytics o Apache. Los esquemas de configuración lo ayudan a aprovisionar canalizaciones fácilmente sin tener que crear una configuración desde cero.

Consola

Cómo usar un esquema de canalización

1. Inicie sesión en la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. Seleccione Canalizaciones en el panel de navegación izquierdo y Seleccione Crear canalización.
3. Seleccione un esquema en la lista de casos de uso y, a continuación, elija Seleccionar esquema. La configuración de la canalización se completa con una subcanalización para el caso de uso que haya seleccionado.

   El esquema de la canalización no es válido tal como está. Debe especificar ajustes adicionales en función del origen seleccionado.

CLI

Para ver una lista de todos los esquemas disponibles mediante la AWS CLI, envíe una solicitud list-pipeline-blueprints.

```
aws osis list-pipeline-blueprints
```

La solicitud devuelve una lista de todos los esquemas disponibles.

Para ver información más detallada sobre un esquema específico, utilice el comando get-pipeline-blueprint:

```
aws osis get-pipeline-blueprint --blueprint-name AWS-ApacheLogPipeline
```

Esta solicitud devuelve el contenido del esquema de la canalización de registro de Apache:

```
{
   "Blueprint":{
      "PipelineConfigurationBody":"###\n  # Limitations: https://docs.aws.amazon.com/
opensearch-service/latest/ingestion/ingestion.html#ingestion-limitations\n###\n###\n
 # apache-log-pipeline:\n    # This pipeline receives logs via http (e.g. FluentBit),
 extracts important values from the logs by matching\n    # the value in the 'log' key
 against the grok common Apache log pattern. The grokked logs are then sent\n    # to
 OpenSearch to an index named 'logs'\n###\n\nversion: \"2\"\napache-log-pipeline:\n
 source:\n    http:\n      # Provide the path for ingestion. ${pipelineName} will be
 replaced with pipeline name configured for this pipeline.\n      # In this case it
 would be \"/apache-log-pipeline/logs\". This will be the FluentBit output URI value.
\n      path: \"/${pipelineName}/logs\"\n  processor:\n    - grok:\n        match:\n
         log: [ \"%{COMMONAPACHELOG_DATATYPED}\" ]\n  sink:\n    - opensearch:\n
   # Provide an AWS OpenSearch Service domain endpoint\n        # hosts: [ \"https://
search-mydomain-1a2a3a4a5a6a7a8a9a0a9a8a7a.us-east-1.es.amazonaws.com\" ]\n        aws:
\n          # Provide the region of the domain.\n          # region: \"us-east-1\"\n
        # Enable the 'serverless' flag if the sink is an Amazon OpenSearch Serverless
 collection\n          # serverless: true\n        index: \"logs\"\n        # Enable
 the S3 DLQ to capture any failed requests in an S3 bucket\n        # dlq:\n
  # s3:\n            # Provide an S3 bucket\n            # bucket: \"your-dlq-bucket-
name\"\n            # Provide a key path prefix for the failed requests\n
  # key_path_prefix: \"${pipelineName}/logs/dlq\"\n            # Provide the region
 of the bucket.\n            # region: \"us-east-1\"\n            # Provide a Role
 ARN with access to the bucket. This role should have a trust relationship with osis-
pipelines.amazonaws.com\n"
      "BlueprintName":"AWS-ApacheLogPipeline"
   }
}
```

API de OpenSearch Ingestion

Para obtener información sobre los esquemas de canalización mediante la API de OpenSearch Ingestion, use las operaciones ListPipelineBlueprints y GetPipelineBlueprint.

Visualización de canalizaciones de Amazon OpenSearch Ingestion

Puede ver los detalles sobre una canalización de Amazon OpenSearch Ingestion mediante la Consola de administración de AWS, la AWS CLI o la API de OpenSearch Ingestion.

Consola

Para ver una canalización

1. Inicie sesión en la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. Seleccione Canalizaciones en el panel de navegación izquierdo.
3. (Opcional) Para ver las canalizaciones con un estado determinado, seleccione Cualquier estado y seleccione un estado para filtrar.

   Una canalización puede tener los siguientes estados:

   - `Active`: la canalización está activa y lista para incorporar datos.
   - `Creating`: la canalización se está creando.
   - `Updating`: la canalización se está actualizando.
   - `Deleting`: la canalización se está eliminando.
   - `Create failed`: la canalización no se pudo crear.
   - `Update failed`: la canalización no se pudo actualizar.
   - `Stop failed`: la canalización no se pudo detener.
   - `Start failed`: la canalización no se pudo iniciar.
   - `Stopping`: la canalización se está deteniendo.
   - `Stopped`: la canalización está detenida y se puede reiniciar en cualquier momento.
   - `Starting`: la canalización se está iniciando.

No se le facturan las OCU de ingesta cuando una canalización está en los estados `Create failed`, `Creating`, `Deleting` y `Stopped`.

CLI

Para ver las canalizaciones mediante la AWS CLI, envíe una solicitud de list-pipelines:

```
aws osis list-pipelines
```

La solicitud devuelve una lista de todas las canalizaciones existentes:

```
{
```

```
    "NextToken": null,
    "Pipelines": [
        {
            "CreatedAt": 1.671055851E9,
            "LastUpdatedAt": 1.671055851E9,
            "MaxUnits": 4,
            "MinUnits": 2,
            "PipelineArn": "arn:aws:osis:us-west-2:123456789012:pipeline/log-pipeline",
            "PipelineName": "log-pipeline",
            "Status": "ACTIVE",
            "StatusReason": {
                "Description": "The pipeline is ready to ingest data."
            }
        },
            "CreatedAt": 1.671055851E9,
            "LastUpdatedAt": 1.671055851E9,
            "MaxUnits": 2,
            "MinUnits": 8,
            "PipelineArn": "arn:aws:osis:us-west-2:123456789012:pipeline/another-
pipeline",
            "PipelineName": "another-pipeline",
            "Status": "CREATING",
            "StatusReason": {
                "Description": "The pipeline is being created. It is not able to ingest
 data."
            }
        }
    ]
}
```

Para ver información sobre una sola canalización, use el comando get-pipeline:

```
aws osis get-pipeline --pipeline-name "my-pipeline"
```

La solicitud devuelve la información de configuración de la canalización especificada:

```
{
    "Pipeline": {
        "PipelineName": "my-pipeline",
        "PipelineArn": "arn:aws:osis:us-east-1:123456789012:pipeline/my-pipeline",
        "MinUnits": 9,
        "MaxUnits": 10,
        "Status": "ACTIVE",
```

```
        "StatusReason": {
            "Description": "The pipeline is ready to ingest data."
        },
        "PipelineConfigurationBody": "log-pipeline:\n source:\n http:\n processor:\n
 - grok:\n match:\nlog: [ '%{COMMONAPACHELOG}' ]\n - date:\n from_time_received: true
\n destination: \"@timestamp\"\n  sink:\n - opensearch:\n hosts: [ \"https://search-
mdp-performance-test-duxkb4qnycd63rpy6svmvyvfpi.us-east-1.es.amazonaws.com\" ]\n index:
 \"apache_logs\"\n aws_sts_role_arn: \"arn:aws:iam::123456789012:role/my-domain-role
\"\n  aws_region: \"us-east-1\"\n  aws_sigv4: true",,
        "CreatedAt": "2022-10-01T15:28:05+00:00",
        "LastUpdatedAt": "2022-10-21T21:41:08+00:00",
        "IngestEndpointUrls": [
            "my-pipeline-123456789012.us-east-1.osis.amazonaws.com"
        ]
    }
}
```


API de OpenSearch Ingestion

Para ver las canalizaciones de ingesta de OpenSearch mediante la API de OpenSearch Ingestion, llame a las operaciones ListPipelines y GetPipeline.

Actualización de las canalizaciones de Amazon OpenSearch Ingestion

Puede actualizar las canalizaciones de Amazon OpenSearch Ingestion mediante la Consola de administración de AWS, la AWS CLI o la API de OpenSearch Ingestion. OpenSearch Ingestion inicia una implementación azul/verde al actualizar la configuración de la canalización. Para obtener más información, consulte the section called “Implementaciones azul/verde para actualizaciones de canalización”.

Temas

- Consideraciones
- Permisos necesarios
- Actualización de las canalizaciones
- Implementaciones azul/verde para actualizaciones de canalización

Consideraciones

Tenga en cuenta lo siguiente cuando actualice una canalización:

- No se puede actualizar el nombre ni la configuración de red de una canalización.
- Si su canalización escribe en un receptor de dominio de VPC, no puede volver atrás y cambiar el receptor a un dominio de VPC diferente una vez creada la canalización. Debe eliminar y volver a crear manualmente la canalización con el nuevo receptor. Aún puede cambiar el receptor de un dominio de VPC a un dominio público, de un dominio público a un dominio de VPC o de un dominio público a otro dominio público.
- Puede cambiar el receptor de canalización en cualquier momento entre un dominio público de OpenSearch Service y una colección de OpenSearch sin servidor.
- Al actualizar la configuración del origen, procesador o receptor de una canalización, OpenSearch Ingestion inicia una implementación azul/verde. Para obtener más información, consulte the section called “Implementaciones azul/verde para actualizaciones de canalización”.
- Al actualizar la configuración del origen, procesador o receptor de una canalización, OpenSearch Ingestion actualiza automáticamente la canalización a la última versión secundaria compatible de la versión principal de Data Prepper que la canalización ejecuta. Este proceso mantiene su canalización actualizada con las últimas correcciones de errores y mejoras de rendimiento.
- Puede seguir actualizando su canalización cuando esté detenida.

Permisos necesarios

OpenSearch Ingestion utiliza los siguientes permisos de IAM para actualizar las canalizaciones:

- `osis:UpdatePipeline`: Actualizar una canalización.
- `osis:ValidatePipeline`: Comprobar si la configuración de la canalización es válida.
- `iam:PassRole`: transfiera el rol de canalización a OpenSearch Ingestion para que pueda escribir datos en el dominio. Este permiso solo es necesario si actualiza la configuración de la canalización, no si modifica otros ajustes, como la publicación de registros o los límites de capacidad.

Por ejemplo, la siguiente política concede permiso para actualizar una canalización:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Resource": "*",
            "Action": [
                "osis:UpdatePipeline",
                "osis:ValidatePipeline"
            ]
        },
        {
            "Resource": [
                "arn:aws:iam::111122223333:role/pipeline-role"
            ],
            "Effect": "Allow",
            "Action": [
                "iam:PassRole"
            ]
        }
    ]
}
```

Actualización de las canalizaciones

Puede actualizar las canalizaciones de Amazon OpenSearch Ingestion mediante la Consola de administración de AWS, la AWS CLI o la API de OpenSearch Ingestion.

Consola

Cómo actualizar una canalización

1. Inicie sesión en la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. Seleccione Canalizaciones en el panel de navegación izquierdo.
3. Seleccione una canalización para abrir su configuración. A continuación, seleccione una de las opciones de edición.

4. Cuando termine de realizar los cambios, seleccione Guardar.

CLI

Para actualizar una canalización mediante el AWS CLI, envíe una solicitud update-pipeline. En el siguiente ejemplo de solicitud, se carga un nuevo archivo de configuración y se actualizan los valores de capacidad mínima y máxima:

```
aws osis update-pipeline \
  --pipeline-name "my-pipeline" \
  --pipline-configuration-body "file://new-pipeline-config.yaml" \
  --min-units 11 \
  --max-units 18
```

API de OpenSearch Ingestion

Para actualizar una canalización de OpenSearch Ingestion mediante la API de OpenSearch Ingestion, llame a la operación UpdatePipeline.

Implementaciones azul/verde para actualizaciones de canalización

OpenSearch Ingestion inicia un proceso de implementación azul/verde al actualizar la configuración de una canalización.

Azul/Verde se refiere a la práctica de crear un nuevo entorno para actualizaciones de canalización y redirigir el tráfico al nuevo entorno una vez completadas dichas actualizaciones. La práctica minimiza el tiempo de inactividad y mantiene el entorno original en caso de que la implementación en el nuevo entorno no se produzca correctamente. Las implementaciones azul/verde en sí mismas no tienen ningún impacto en el rendimiento, pero este puede cambiar si la configuración de la canalización cambia de una manera que altere el rendimiento.

OpenSearch Ingestion bloquea el escalado automático durante las implementaciones azul/verde. Se le seguirá cobrando solo por el tráfico a la canalización anterior hasta que se redirija a la nueva canalización. Una vez que se haya redirigido el tráfico, solo se cobrará por la nueva canalización. Nunca se le cobrará por dos canalizaciones simultáneamente.

Al actualizar la configuración del origen, procesador o receptor de una canalización, OpenSearch Ingestion puede actualizar automáticamente la canalización a la última versión secundaria compatible de la versión principal que la canalización ejecuta. Por ejemplo, es posible que tenga `version: "2"` en la configuración de su canalización y OpenSearch Ingestion aprovisionó

inicialmente la canalización con la versión 2.1.0. Cuando se añade la compatibilidad con la versión 2.1.1 y realiza un cambio en la configuración de la canalización, OpenSearch Ingestion la actualiza a la versión 2.1.1.

Este proceso mantiene su canalización actualizada con las últimas correcciones de errores y mejoras de rendimiento. OpenSearch Ingestion no puede actualizar la versión principal de su canalización a menos que cambie manualmente la opción `version`en la configuración de la canalización.

Administración de los costos de las canalizaciones de Amazon OpenSearch Ingestion

Puede iniciar y detener los procesos de ingesta en Amazon OpenSearch Ingestion para controlar el flujo de datos en función de sus necesidades. Al detener una canalización, se detiene el procesamiento de datos y, al mismo tiempo, se conservan las configuraciones, por lo que puede reiniciarla sin tener que volver a configurarla. Esto puede ayudar a optimizar los costos, administrar el uso de los recursos o solucionar problemas. Al detener una canalización, OpenSearch Ingestion no procesa los datos entrantes, pero los datos ingeridos anteriormente permanecen disponibles en OpenSearch.

El inicio y la detención simplifican los procesos de configuración y eliminación de canalizaciones que usa para desarrollo, pruebas o actividades similares que no requieren disponibilidad continua. Mientras su canalización esté detenida, no se le cobrará ninguna hora de Ingestion OCU. Puede seguir actualizando las canalizaciones detenidas y estas reciben actualizaciones automáticas de las versiones secundarias y revisiones de seguridad. Al reiniciar una canalización, se reanuda el procesamiento a partir de los nuevos datos entrantes.

> **Note**
>
> Si su canalización tiene un exceso de capacidad pero necesita permanecer operativa, considera ajustar sus límites de capacidad máxima en lugar de detenerla y reiniciarla. Esto puede ayudar a administrar los costos y, al mismo tiempo, garantizar que la canalización continúe procesando los datos de manera eficiente. Para obtener más información, consulta the section called “Escala de canalizaciones”.

Los siguientes temas explican cómo iniciar y detener canalizaciones utilizando la API de Consola de administración de AWS, AWS CLI y OpenSearch Ingestion.

Temas

- Detención de una canalización de Amazon OpenSearch Ingestion
- Inicio de una canalización de Amazon OpenSearch Ingestion

Detención de una canalización de Amazon OpenSearch Ingestion

Para usar una canalización de OpenSearch Ingestion o realizar tareas de administración, siempre se empieza con una canalización activa, después se detiene la canalización y se vuelve a iniciar. Mientras su canalización esté detenida, no se le cobrará ninguna hora de Ingestion OCU.

Consola

Cómo detener una canalización

1. Inicie sesión en la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. En el panel de navegación, seleccione Canalizaciones, y luego seleccione una canalización. Puede realizar la operación de detención desde esta página o navegar a la página de detalles de la canalización de datos que desea detener.
3. En Acciones, seleccione Detener canalización.

   Si no se puede detener ni iniciar una canalización, la acción Detener canalización no estará disponible.

AWS CLI

Para detener una canalización mediante AWS CLI, llame al comando stop-pipeline con los siguientes parámetros:

- `--pipeline-name`: el nombre de la canalización.

Example

```
aws osis stop-pipeline --pipeline-name my-pipeline
```

API de OpenSearch Ingestion

Para detener una canalización con la API de OpenSearch Ingestion, llame a la operación StopPipeline con el siguiente parámetro:

- `PipelineName`: el nombre de la canalización.

Inicio de una canalización de Amazon OpenSearch Ingestion

Para iniciar una canalización de OpenSearch Ingestion, siempre debe comenzar con una canalización que ya está en estado detenido. La canalización mantiene sus ajustes de configuración como límites de capacidad, configuración de red y opciones de publicación de registros.

El reinicio de la canalización suele tardar varios minutos.

Consola

Cómo iniciar una canalización

1. Inicie sesión en la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. En el panel de navegación, seleccione Canalizaciones, y luego seleccione una canalización. Puede realizar la operación de inicio desde esta página o navegar a la página de detalles de la canalización que desea iniciar.
3. En Acciones, seleccione Iniciar canalización.

AWS CLI

Para iniciar una canalización con la AWS CLI, llame al comando start-pipeline con los siguientes parámetros:

- `--pipeline-name`: el nombre de la canalización.

Example

```
aws osis start-pipeline --pipeline-name my-pipeline
```

API de OpenSearch Ingestion

Para iniciar una canalización de OpenSearch Ingestion con la API de OpenSearch Ingestion, llame a la operación StartPipeline con el siguiente parámetro:

- `PipelineName`: el nombre de la canalización.

Eliminación de canalizaciones de Amazon OpenSearch Ingestion

Puede eliminar una canalización de Amazon OpenSearch Ingestion mediante laConsola de administración de AWS, la AWS CLI o la API de OpenSearch Ingestion. No puede eliminar una canalización si tiene un estado de `Creating` o `Updating`.

Consola

Cómo eliminar una canalización

1. Inicie sesión en la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. Seleccione Canalizaciones en el panel de navegación izquierdo.
3. Seleccione la canalización que desee eliminar y seleccione Acciones, Eliminar.
4. Confirme la eliminación y elija Eliminar.

CLI

Para eliminar una canalización mediante AWS CLI, envíe una solicitud de delete-pipeline:

```
aws osis delete-pipeline --pipeline-name "my-pipeline"
```

API de OpenSearch Ingestion

Para eliminar una canalización de OpenSearch Ingestion mediante la API de OpenSearch Ingestion, llame a la operación DeletePipeline con el siguiente parámetro:

- `PipelineName`: el nombre de la canalización.

Complementos y opciones compatibles para las canalizaciones de Amazon OpenSearch Ingestion

Amazon OpenSearch Ingestion admite un subconjunto de fuentes, procesadores y destinos dentro de OpenSearch Data Prepper de código abierto. Además, OpenSearch Ingestion impone algunas restricciones a las opciones disponibles para cada complemento compatible. En las siguientes secciones se describen los complementos y las opciones asociadas que admite OpenSearch Ingestion.

> **Note**
>
> OpenSearch Ingestion no admite ningún complemento de búfer porque configura automáticamente un búfer predeterminado. Si incluye un búfer en la configuración de la canalización, recibirá un error de validación.

Temas

- Complementos compatibles
- Procesadores sin estado frente a procesadores con estado
- Requisitos y restricciones de configuración

Complementos compatibles

OpenSearch Ingestion es compatible con los siguientes complementos de Data Prepper:

Fuentes:

- DocumentDB
- DynamoDB

- HTTP
- Kafka
- Kinesis
- OpenSearch
- Registros de OTel

- Métricas de OTel
- Rastreo de OTel
- S3

Procesadores:

- Agregar entradas
- Agregado
- Detector de anomalías
- AWS Lambda
- Convertir el tipo de entrada
- Copiar valores
- CSV
- Date
- Delay
- Descomprimir
- Eliminar entradas
- Diseccionar
- Soltar eventos
- Flatten
- IP geográfica
- Grok
- Valor de clave
- Lista a mapa
- Cadena en minúsculas
- Mapa a lista
- Evento de mutación (serie de procesadores)
- Cadena de mutación (serie de procesadores)
- Obfuscate
- Métricas de OTel

- Grupo de rastreo OTel
- Rastreo de OTel
- Analizar Ion
- Analizar JSON
- Analizar XML
- Cambiar el nombre de las claves
- Seleccionar entradas
- Mapa de servicios
- Evento dividido
- Cadena dividida
- Convertidor de cadenas
- Cadena sustitutiva
- Reenviador por pares de seguimiento
- Translate
- Recortar cadena
- Truncar
- Cadena en mayúscula
- Agente de usuario
- Escribir JSON

Receptores:

- OpenSearch (compatible con OpenSearch Service, OpenSearch sin servidor y Elasticsearch 6.8 o versiones posteriores)
- S3

Códecs de receptor:

- Avro
- NDJSON
- JSON
- Parquet

Procesadores sin estado frente a procesadores con estado

Los procesadores sin estado llevan a cabo operaciones como las transformaciones y el filtrado, mientras que los procesadores con estado llevan a cabo operaciones como las agregaciones que recuerdan el resultado de la ejecución anterior. OpenSearch Ingestion es compatible con los procesadores con estado Aggregate y Service-Map. Todos los demás procesadores compatibles no tienen estado.

En el caso de las canalizaciones que solo contienen procesadores sin estado, el límite máximo de capacidad es de 96 OCU de ingesta. Si la canalización contiene algún procesador con estado, el límite máximo de capacidad es de 48 OCU de ingesta. Sin embargo, si una canalización tiene habilitado el almacenamiento en búfer persistente, puede tener un máximo de 384 OCU de ingesta solo con procesadores sin estado, o 192 OCU de ingesta si contiene algún procesador con estado. Para obtener más información, consulte the section called “Escala de canalizaciones”.

El reconocimiento de extremo a extremo solo es compatible con procesadores sin estado. Para más información, consulte the section called “Reconocimiento integral”.

Requisitos y restricciones de configuración

A menos que se especifique lo contrario a continuación, todas las opciones descritas en la referencia de configuración de Data Prepper para los complementos compatibles enumerados anteriormente están permitidas en las canalizaciones de OpenSearch Ingestion. En las siguientes secciones se explican las restricciones que OpenSearch Ingestion impone a determinadas opciones de los complementos.

> **Note**
>
> OpenSearch Ingestion no admite ningún complemento de búfer porque configura automáticamente un búfer predeterminado. Si incluye un búfer en la configuración de la canalización, recibirá un error de validación.

OpenSearch Ingestion configura y administra internamente muchas opciones, como `authentication` y `acm_certificate_arn`. Otras opciones, como `thread_count` y `request_timeout`, tienen un impacto en el rendimiento si se modifican manualmente. Por lo tanto, estos valores se establecen internamente para garantizar un rendimiento óptimo de sus canalizaciones.

Por último, algunas opciones no se pueden pasar a OpenSearch Ingestion, como `ism_policy_file` y `sink_template`, porque son archivos locales cuando se ejecutan en Data Prepper de código abierto. No se admiten estos valores.

Temas

- Opciones generales de canalización
- Procesador Grok
- Origen de HTTP
- Receptor de OpenSearch
- Fuente de métricas de OTel, fuente de rastreo de OTel y fuente de registros de OTel
- Procesador de grupos de rastreo de OTel
- Procesador de rastreo de OTel
- Procesador de mapas de servicio
- Fuente de S3

Opciones generales de canalización

OpenSearch Ingestion establece las siguientes opciones generales de canalización que no se admiten en las configuraciones de canalización:

- `workers`
- `delay`

Procesador Grok

No se admiten las siguientes opciones de procesador Grok:

- `patterns_directories`
- `patterns_files_glob`

Origen de HTTP

El complemento fuente HTTP tiene los siguientes requisitos y restricciones:

- La opción `path` es obligatoria. La ruta es una cadena, por ejemplo `/log/ingest`, que representa la ruta del URI para la ingesta de registros. Esta ruta define el URI que usa para enviar datos a la canalización. Por ejemplo, `https://log-pipeline.us-west-2.osis.amazonaws.com/log/ingest`. La ruta debe empezar con una barra (/) y puede contener los caracteres especiales '-', '_', '.' y '/', así como el marcador de posición `${pipelineName}`.
- OpenSearch Ingestion establece las siguientes opciones de origen de HTTP que no se admiten en las configuraciones de canalización:
  - `port`
  - `ssl`
  - `ssl_key_file`
  - `ssl_certificate_file`
  - `aws_region`
  - `authentication`
  - `unauthenticated_health_check`
  - `use_acm_certificate_for_ssl`
  - `thread_count`
  - `request_timeout`
  - `max_connection_count`
  - `max_pending_requests`
  - `health_check_service`
  - `acm_private_key_password`
  - `acm_certificate_timeout_millis`
  - `acm_certificate_arn`

Receptor de OpenSearch

El complemento del receptor de OpenSearch tiene los siguientes requisitos y limitaciones.

- La opción `aws` es obligatoria y debe contener las siguientes opciones:
  - `sts_role_arn`
  - `region`
  - `hosts`

  - `serverless` (si el receptor es una colección OpenSearch sin servidor)
- La opción `sts_role_arn` debe apuntar al mismo rol para cada receptor de un archivo de definición YAML.
- La opción `hosts` debe especificar un punto de conexión del dominio de OpenSearch Service o un punto de conexión de la colección de OpenSearch sin servidor. No se puede especificar un punto de conexión personalizado para un dominio; este debe ser el punto de conexión estándar.
- Si la opción de `hosts` es un punto de conexión de la colección sin servidor, debe configurar la opción `serverless` en `true`. Además, si su archivo de definición YAML contiene la opción `index_type`, debe estar configurada en `management_disabled`, de lo contrario, la validación fallará.
- Las siguientes opciones no son compatibles:
  - `username`
  - `password`
  - `cert`
  - `proxy`
  - `dlq_file`: si quiere descargar los eventos fallidos a una cola de mensajes fallidos (DLQ), debe usar la opción `dlq` y especificar un bucket de S3.
  - `ism_policy_file`
  - `socket_timeout`
  - `template_file`
  - `insecure`

Fuente de métricas de OTel, fuente de rastreo de OTel y fuente de registros de OTel

Los complementos de la fuente de métricas de OTel, la fuente de rastreo de OTel y los registros de OTel tienen los siguientes requisitos y limitaciones:

- La opción `path` es obligatoria. La ruta es una cadena, por ejemplo `/log/ingest`, que representa la ruta del URI para la ingesta de registros. Esta ruta define el URI que usa para enviar datos a la canalización. Por ejemplo, `https://log-pipeline.us-west-2.osis.amazonaws.com/log/ingest`. La ruta debe empezar con una barra (/) y puede contener los caracteres especiales '-', '_', '.' y '/', así como el marcador de posición `${pipelineName}`.
- OpenSearch Ingestion establece las siguientes opciones que no se admiten en las configuraciones de canalización:

- `port`
- `ssl`
- `sslKeyFile`
- `sslKeyCertChainFile`
- `authentication`
- `unauthenticated_health_check`
- `useAcmCertForSSL`
- `unframed_requests`
- `proto_reflection_service`
- `thread_count`
- `request_timeout`
- `max_connection_count`
- `acmPrivateKeyPassword`
- `acmCertIssueTimeOutMillis`
- `health_check_service`
- `acmCertificateArn`
- `awsRegion`

Procesador de grupos de rastreo de OTel

El procesador de grupos de rastreo de OTel tiene los siguientes requisitos y limitaciones:

- La opción `aws` es obligatoria y debe contener las siguientes opciones:
  - `sts_role_arn`
  - `region`
  - `hosts`
- La `sts_role_arn` opción especifica el mismo rol que el rol de canalización que se especifica en la configuración del receptor de OpenSearch.
- No se admiten las opciones `username`, `password`, `cert`, y `insecure`.
- La opción `aws_sigv4` es obligatoria y se debe establecer en verdadero.

- La opción `serverless` del complemento del receptor de OpenSearch no es compatible. El procesador del grupo de rastreo de OTel no funciona actualmente con los receptores de las colecciones de OpenSearch sin servidor.
- El número de procesadores `otel_trace_group` en el cuerpo de configuración de la canalización no puede ser superior a 8.

Procesador de rastreo de OTel

El procesador de rastreo de OTel tiene los siguientes requisitos y limitaciones:

- El valor de la opción `trace_flush_interval` no puede superar los 300 segundos.

Procesador de mapas de servicio

El procesador de Service-map tiene los siguientes requisitos y limitaciones:

- El valor de la opción `window_duration` no puede superar los 300 segundos.

Fuente de S3

El complemento fuente de S3 tiene los siguientes requisitos y limitaciones:

- La opción `aws` es obligatoria y debe contener las opciones `region` y `sts_role_arn`.
- El valor de la opción `records_to_accumulate` no puede ser superior a 200.
- El valor de la opción `maximum_messages` no puede ser superior a 10.
- Si se especifica, la opción `disable_bucket_ownership_validation` se debe establecer en falso.
- Si se especifica, la opción `input_serialization` se debe establecer en `parquet`.

Integración de canalizaciones de Amazon OpenSearch Ingestion con otros servicios y aplicaciones

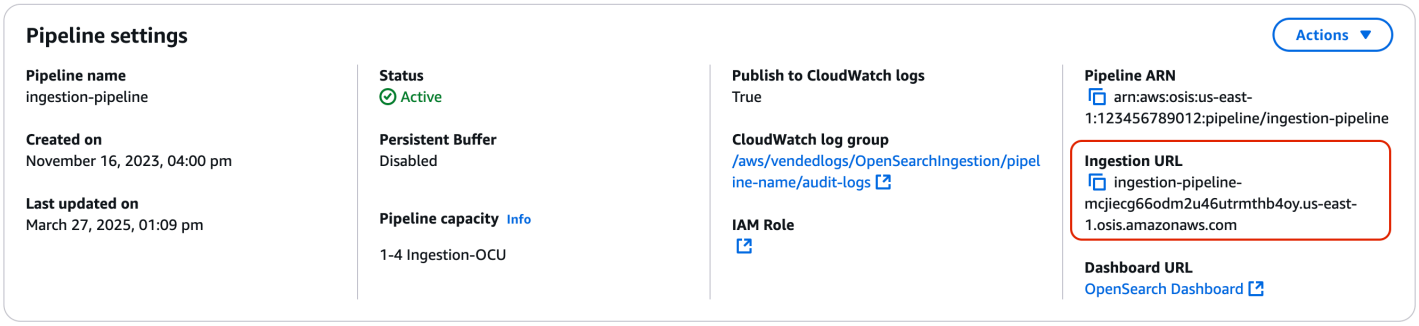
Para poder ingerir datos correctamente en una canalización de Amazon OpenSearch Ingestion, debe configurar la aplicación cliente (el origen) para que envíe los datos al punto de conexión de

la canalización. Su fuente pueden ser clientes como los registros de Fluent Bit, OpenTelemetry Collector o un bucket simple de S3. La configuración exacta es diferente para cada cliente.

Las diferencias importantes durante la configuración de la fuente (en comparación con el envío de datos directamente a un dominio de OpenSearch Service o a una colección de OpenSearch sin servidor) son el nombre del servicio de AWS (`osis`) y el punto de conexión del host, que debe ser el punto de conexión de la canalización.

Creación del punto de conexión de ingesta

Para ingerir datos a una canalización, envíelos al punto de conexión de ingesta. Para localizar la URL de ingesta, vaya a la página Configuración de canalización y copie la URL de ingesta.



Para crear el punto de conexión de ingesta completo para fuentes basadas en pull, como el rastreo OTel y las métricas de OTel, agregue la ruta de ingesta desde la configuración de la canalización hasta la URL de ingesta.

Por ejemplo, supongamos que su configuración de canalización tiene la siguiente ruta de ingesta:



El punto de conexión de ingesta completo, que se especifica en la configuración de su cliente, tendrá el siguiente formato: `https://`*`ingestion-pipeline-abcdefg`*`.us-east-1.osis.amazonaws.com`*`/my/test_path`*.

Creación de un rol de ingesta

Todas las solicitudes para OpenSearch Ingestion deben estar firmadas con Signature Version 4. Como mínimo, el rol que firma la solicitud debe tener permiso para realizar la acción de `osis:Ingest`, lo que le permitirá enviar datos a una canalización de OpenSearch Ingestion.

Por ejemplo, la siguiente política de AWS Identity and Access Management (IAM) permite que el rol correspondiente envíe datos a una única canalización:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "osis:Ingest",
      "Resource": "arn:aws:osis:us-east-1:111122223333:pipeline/pipeline-name"
    }
  ]
}
```

> **Note**
>
> Para usar el rol en todas las canalizaciones, sustituya el ARN del elemento `Resource` por un comodín (*).

Provisión de acceso de ingesta entre cuentas

> **Note**
>
> Solo puede proporcionar acceso de ingesta entre cuentas a las canalizaciones públicas, no a las canalizaciones de VPC.

Es posible que tenga que incorporar datos en una canalización desde una Cuenta de AWS diferente, como una cuenta que aloja su aplicación de origen. Si la entidad principal que escribe en una

canalización está en una cuenta diferente a la de la propia canalización, debe configurar la entidad principal para que confíe en otro rol de IAM para incorporar los datos a la canalización.

Cómo configurar los permisos de ingesta entre cuentas

1. Cree el rol de ingesta con el permiso `osis:Ingest` (descrito en la sección anterior) dentro de la misma Cuenta de AWS que la canalización. Para ver instrucciones, consulte Creación de roles de IAM.
2. Adjunte una política de confianza al rol de ingesta que permita que la entidad principal de otra cuenta lo asuma:

   JSON

   ```
   {
     "Version":"2012-10-17",
     "Statement": [{
        "Effect": "Allow",
        "Principal": {
          "AWS": "arn:aws:iam::111122223333:root"
         },
        "Action": "sts:AssumeRole"
     }]
   }
   ```

3. En la otra cuenta, configure la aplicación cliente (por ejemplo, Fluent Bit) para que asuma el rol de ingesta. Para que esto funcione, la cuenta de la aplicación debe conceder permisos al usuario o rol de la aplicación para que asuma el rol de ingesta.

   El siguiente ejemplo de política basada en identidad permite que la entidad principal adjunta asuma el `ingestion-role` de la cuenta de canalización:

   JSON

   ```
   {
     "Version":"2012-10-17",
     "Statement": [
       {
         "Effect": "Allow",
         "Action": "sts:AssumeRole",
         "Resource": "arn:aws:iam::111122223333:role/ingestion-role"
   ```

```
    }
  ]
}
```

Luego, la aplicación cliente puede utilizar la operación AssumeRole para asumir el `ingestion-role` e incorporar datos en la canalización asociada.

Uso de una canalización de OpenSearch Ingestion con Atlassian Services

Puede utilizar los complementos de origen Atlassian Jira y Confluence para ingerir datos de los servicios de Atlassian en su canalización de OpenSearch Ingestion. Estas integraciones le permiten crear una base de conocimientos unificada con capacidad de búsqueda al sincronizar proyectos completos de Jira y los espacios de Confluence y, al mismo tiempo, mantener la relevancia en tiempo real mediante la supervisión continua y la sincronización automática de las actualizaciones.

Integrating with Jira

Transforme su experiencia con Jira con potentes funciones de búsqueda contextual integrando su contenido de Jira en OpenSearch. El complemento de origen Atlassian Jira de Data Prepper le permite crear una base de conocimientos unificada y con capacidad de búsqueda mediante la sincronización de proyectos completos de Jira, al tiempo que mantiene la relevancia en tiempo real gracias a la supervisión continua y la sincronización automática de las actualizaciones. Esta integración permite la sincronización de datos con opciones de filtrado flexibles para proyectos, tipos de problemas y estados específicos, lo que garantiza que solo se importe la información que necesita.

Para garantizar una conectividad segura y confiable, el complemento admite varios métodos de autenticación, incluida la autenticación básica con clave de API y la autenticación OAuth2, con la seguridad adicional de administrar las credenciales mediante un secreto almacenado en AWS Secrets Manager. También cuenta con la renovación automática de los tokens para un acceso ininterrumpido, lo que garantiza un funcionamiento continuo. Basada en la versión 2 de la API de Atlassian, esta integración permite a los equipos obtener información valiosa a partir de sus datos de Jira mediante las funciones de búsqueda avanzada de OpenSearch.

Integrating with Confluence

Mejore las capacidades de administración del conocimiento y colaboración de su equipo con la integración de contenido de Atlassian Confluence en OpenSearch a través del complemento de origen Confluence de Data Prepper. Esta integración le permite crear un repositorio de

conocimiento colectivo centralizado y con capacidad de búsqueda, lo que mejora la detección de información y la productividad del equipo. Al sincronizar el contenido de Confluence y monitorear continuamente las actualizaciones, el complemento garantiza que su índice de OpenSearch permanezca actualizado y completo.

La integración ofrece opciones de filtrado flexibles, lo que le permite importar contenido de forma selectiva desde espacios o tipos de páginas específicos, adaptando el contenido sincronizado a las necesidades de su organización. El complemento admite tanto la clave de API básica como los métodos de autenticación de OAuth2, con la opción de administrar las credenciales de forma segura mediante AWS Secrets Manager. La característica de renovación automática de los tokens del complemento garantiza un acceso ininterrumpido y un funcionamiento sin problemas. Basada en la API Confluence de Atlassian, esta integración permite a los equipos aprovechar las capacidades de búsqueda avanzada de OpenSearch en todo su contenido de Confluence, lo que mejora la accesibilidad y la utilización de la información dentro de la organización.

Temas

- Requisitos previos
- Configurar un rol de canalización
- Configuración de una canalización de Jira Connector
- Configuración de una canalización del conector de Confluence
- Coherencia de datos
- Limitaciones
- Métricas en CloudWatch para conectores de Atlassian
- Conexión de una canalización de Amazon OpenSearch Ingestion a Atlassian Jira o Confluence mediante OAuth 2.0

Requisitos previos

Antes de crear su canalización de OpenSearch Ingestion, complete los siguientes pasos:

1. Prepare credenciales para su sitio de Jira eligiendo una de las siguientes opciones. OpenSearch Ingestion solo requiere la autorización `ReadOnly` del contenido.
   a. Opción 1: clave de API: inicie sesión en su cuenta de Atlassian y utilice la información del siguiente tema para generar su clave de API:
      - Gestione los tokens de API para su cuenta de Atlassian

b. Opción 2: OAuth2: inicie sesión en su cuenta de Atlassian y utilice la información en the section called “Conexión de una canalización de Amazon OpenSearch Ingestion a Atlassian Jira o Confluence mediante OAuth 2.0”.

2. Cree un secreto de AWS Secrets Manager para almacenar los detalles de las credenciales que creó en el paso anterior: Realice las siguientes elecciones a medida que sigue el procedimiento:
   - En Secret type (Tipo de secreto), elija Other type of secret (Otro tipo de secreto).
   - Para los pares clave/valor, cree los siguientes pares, según el tipo de autorización seleccionado:

   API key

   ```
   {
      "username": user-name-usualy-email-id,
      "password": api-key
   }
   ```

   OAuth 2.0

   ```
   {
      "clientId": client-id
      "clientSecret": client-secret
      "accessKey": access-key
      "refreshKey": refresh-key
   }
   ```

   Una vez creado el secreto, copie el nombre de recurso de Amazon (ARN) del secreto. Lo incluirá en la política de permisos de los roles en proceso.

Configurar un rol de canalización

El rol transferido en la canalización debe incluir la siguiente política de lectura y escritura en el secreto creado en la sección de requisitos previos.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "SecretReadWrite",
```

```
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetResourcePolicy",
                "secretsmanager:GetSecretValue",
                "secretsmanager:DescribeSecret",
                "secretsmanager:PutSecretValue",
                "secretsmanager:ListSecretVersionIds"
            ],
            "Resource": "arn:aws:secretsmanager:us-
east-1:111122223333:secret:secret-name-random-6-characters"
        }
    ]
}
```

El rol también debe incluir una política de acceso y escritura en el servidor que elijas. Por ejemplo, si elige OpenSearch como receptor, la política tendrá un aspecto similar al siguiente:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "OpenSearchWritePolicy",
            "Effect": "Allow",
            "Action": "aoss:*",
            "Resource": "arn:aws:aoss:us-
east-1:111122223333:collection/collection-id"
        }
    ]
}
```

Configuración de una canalización de Jira Connector

Puede utilizar un esquema de Atlassian Jira preconfigurado para crear esta canalización. Para obtener más información, consulte the section called “Uso de esquemas”.

Reemplace los *valores de marcador de posición* con su información.

```
version: "2"
```

```
extension:
  aws:
    secrets:
      jira-account-credentials:
        secret_id: "secret-arn"
        region: "secret-region"
        sts_role_arn: "arn:aws:iam::123456789012:role/Example-Role"
atlassian-jira-pipeline:
  source:
    jira:
      # We only support one host url for now
      hosts: ["jira-host-url"]
      acknowledgments: true
      authentication:
        # Provide one of the authentication method to use. Supported methods are
 'basic' and 'oauth2'.
        # For basic authentication, password is the API key that you generate using
 your jira account
        basic:
          username: ${{aws_secrets:jira-account-credentials:username}}
          password: ${{aws_secrets:jira-account-credentials:password}}
        # For OAuth2 based authentication, we require the following 4 key values stored
 in the secret
        # Follow atlassian instructions at the below link to generate these keys.
        # https://developer.atlassian.com/cloud/confluence/oauth-2-3lo-apps/
        # If you are using OAuth2 authentication, we also require, write permission to
 your AWS secret to
        # be able to write the renewed tokens back into the secret.
        # oauth2:
          # client_id: ${{aws_secrets:jira-account-credentials:clientId}}
          # client_secret: ${{aws_secrets:jira-account-credentials:clientSecret}}
          # access_token: ${{aws_secrets:jira-account-credentials:accessToken}}
          # refresh_token: ${{aws_secrets:jira-account-credentials:refreshToken}}
      filter:
        project:
          key:
            include:
              # This is not project name.
              # It is an alphanumeric project key that you can find under project
 details in Jira.
              - "project-key"
              - "project-key"
            # exclude:
              # - "project-key"
```

```
              # - "project-key"
        issue_type:
          include:
            - "issue-type"
            # - "Story"
            # - "Bug"
            # - "Task"
         # exclude:
             # - "Epic"
        status:
          include:
            - "ticket-status"
            # - "To Do"
            # - "In Progress"
            # - "Done"
         # exclude:
           # - "Backlog"

  sink:
    - opensearch:
        # Provide an Amazon OpenSearch Service domain endpoint
        hosts: [ "https://search-mydomain-1a2a3a4a5a6a7a8a9a0a9a8a7a.us-
east-1.es.amazonaws.com" ]
        index: "index_${getMetadata(\"project\")}"
        # Ensure adding unique document id which is the unique ticket id in this case
        document_id: '${/id}'
        aws:
          # Provide a Role ARN with access to the domain. This role should have a trust
 relationship with osis-pipelines.amazonaws.com
          sts_role_arn: "arn:aws:iam::123456789012:role/Example-Role"
          # Provide the region of the domain.
          region: "us-east-1"
          # Enable the 'serverless' flag if the sink is an Amazon OpenSearch Serverless
 collection
          serverless: false
          # serverless_options:
            # Specify a name here to create or update network policy for the serverless
 collection
            # network_policy_name: "network-policy-name"
        # Enable the 'distribution_version' setting if the Amazon OpenSearch Service
 domain is of version Elasticsearch 6.x
        # distribution_version: "es6"
        # Enable and switch the 'enable_request_compression' flag if the default
 compression setting is changed in the domain.
```

```
        # See the section called "Compresión de solicitudes HTTP"
        # enable_request_compression: true/false
        # Optional: Enable the S3 DLQ to capture any failed requests in an S3 bucket.
Delete this entire block if you don't want a DLQ.
        dlq:
          s3:
            # Provide an S3 bucket
            bucket: "your-dlq-bucket-name"
            # Provide a key path prefix for the failed requests
            # key_path_prefix: "kinesis-pipeline/logs/dlq"
            # Provide the region of the bucket.
            region: "us-east-1"
            # Provide a Role ARN with access to the bucket. This role should have a
trust relationship with osis-pipelines.amazonaws.com
            sts_role_arn: "arn:aws:iam::123456789012:role/Example-Role"
```

Atributos clave del origen de Jira:

1. hosts: la URL en las instalaciones o en la nube de Jira. En general, se ve similar a `https://your-domain-name.atlassian.net/`.
2. acknowledgments: para garantizar la entrega de los datos hasta el receptor.
3. authentication: describe cómo desea que la canalización acceda a su instancia de Jira. Elija `Basic` o `OAuth2` y especifique los atributos clave correspondientes que hagan referencia a las claves de su secreto de AWS.
4. filter: esta sección lo ayuda a seleccionar qué parte de sus datos de Jira desea extraer y sincronizar.
   a. project: enumere las claves del proyecto que desea sincronizar en la sección `include`. De lo contrario, enumere los proyectos que desea excluir en la sección `exclude`. Proporcione solo una de las opciones de inclusión o exclusión en un momento dado.
   b. issue_type: tipos de problemas específicos que desea sincronizar. Siga un patrón similar a `include` o `exclude` que se adapte a sus necesidades. Tenga en cuenta que los archivos adjuntos aparecerán como enlaces de anclaje al archivo adjunto original, pero el contenido del archivo adjunto no se extraerá.
   c. status: filtro de estado específico que desea aplicar a la consulta de extracción de datos. Si especifica `include`, solo se sincronizarán los tickets con esos estados. Si especifica `exclude`, se sincronizarán todos los tickets, excepto los que tengan los estados de exclusión enumerados.

Configuración de una canalización del conector de Confluence

Puede utilizar un esquema de Atlassian Confluence preconfigurado para crear esta canalización. Para obtener más información, consulte the section called “Uso de esquemas”.

```
version: "2"
extension:
  aws:
    secrets:
      confluence-account-credentials:
        secret_id: "secret-arn"
        region: "secret-region"
        sts_role_arn: "arn:aws:iam::123456789012:role/Example-Role"
atlassian-confluence-pipeline:
  source:
    confluence:
      # We currently support only one host URL.
      hosts: ["confluence-host-url"]
      acknowledgments: true
      authentication:
        # Provide one of the authentication method to use. Supported methods are
 'basic' and 'oauth2'.
        # For basic authentication, password is the API key that you generate using
 your Confluence account
        basic:
          username: ${{aws_secrets:confluence-account-credentials:confluenceId}}
          password: ${{aws_secrets:confluence-account-
credentials:confluenceCredential}}
        # For OAuth2 based authentication, we require the following 4 key values stored
 in the secret
        # Follow atlassian instructions at the following link to generate these keys:
        # https://developer.atlassian.com/cloud/confluence/oauth-2-3lo-apps/
        # If you are using OAuth2 authentication, we also require write permission to
 your AWS secret to
        # be able to write the renewed tokens back into the secret.
        # oauth2:
          # client_id: ${{aws_secrets:confluence-account-credentials:clientId}}
          # client_secret: ${{aws_secrets:confluence-account-credentials:clientSecret}}
          # access_token: ${{aws_secrets:confluence-account-credentials:accessToken}}
          # refresh_token: ${{aws_secrets:confluence-account-credentials:refreshToken}}
      filter:
        space:
          key:
```

```
            include:
              # This is not space name.
              # It is a space key that you can find under space details in Confluence.
              - "space key"
              - "space key"
            # exclude:
              #  - "space key"
              #  - "space key"
        page_type:
          include:
            - "content type"
            # - "page"
            # - "blogpost"
            # - "comment"
         # exclude:
            # - "attachment"

  sink:
    - opensearch:
        # Provide an Amazon OpenSearch Service domain endpoint
        hosts: [ "https://search-mydomain-1a2a3a4a5a6a7a8a9a0a9a8a7a.us-
east-1.es.amazonaws.com" ]
         index: "index_${getMetadata(\"space\")}"
        # Ensure adding unique document id which is the unique ticket ID in this case.
        document_id: '${/id}'
        aws:
          # Provide the Amazon Resource Name (ARN) for a role with access to the
 domain. This role should have a trust relationship with osis-pipelines.amazonaws.com.
          sts_role_arn: "arn:aws:iam::123456789012:role/Example-Role"
          # Provide the Region of the domain.
          region: "us-east-1"
          # Enable the 'serverless' flag if the sink is an Amazon OpenSearch Serverless
 collection
          serverless: false
          # serverless_options:
            # Specify a name here to create or update network policy for the serverless
 collection.
            # network_policy_name: "network-policy-name"
        # Enable the 'distribution_version' setting if the Amazon OpenSearch Service
 domain is of version Elasticsearch 6.x
        # distribution_version: "es6"
        # Enable and switch the 'enable_request_compression' flag if the default
 compression setting is changed in the domain.
```

```
        # For more information, see the section called “Compresión de solicitudes
 HTTP”.
        # enable_request_compression: true/false
        # Optional: Enable the S3 DLQ to capture any failed requests in an S3 bucket.
 Delete this entire block if you don't want a DLQ.
        dlq:
          s3:
            # Provide an S3 bucket
            bucket: "your-dlq-bucket-name"
            # Provide a key path prefix for the failed requests
            # key_path_prefix: "kinesis-pipeline/logs/dlq"
            # Provide the Rregion of the bucket.
            region: "us-east-1"
            # Provide the Amazon Resource Name (ARN) for a role with access to the
 bucket. This role should have a trust relationship with osis-pipelines.amazonaws.com
            sts_role_arn: "arn:aws:iam::123456789012:role/Example-Role"
```

Atributos clave del origen de Confluence:

1. hosts: la URL en las instalaciones o en la nube de Confluence. En general, se ve similar a `https://your-domain-name.atlassian.net/`.
2. acknowledgments: para garantizar la entrega de los datos hasta el receptor.
3. authentication: describe cómo desea que la canalización acceda a su instancia de Confluence. Elija `Basic` o `OAuth2` y especifique los atributos clave correspondientes que hagan referencia a las claves de su secreto de AWS.
4. filter: esta sección lo ayuda a seleccionar qué parte de sus datos de Confluence desea extraer y sincronizar.
   a. space: enumere las claves de espacio que desea sincronizar en la sección `include`. De lo contrario, enumere los espacios que desea excluir en la sección `exclude`. Proporcione solo una de las opciones de inclusión o exclusión en un momento dado.
   b. page_type: tipos de página específicos (como página, entrada de blog o archivos adjuntos) que desea sincronizar. Siga un patrón similar a `include` o `exclude` que se adapte a sus necesidades. Tenga en cuenta que los archivos adjuntos aparecerán como enlaces de anclaje al archivo adjunto original, pero el contenido del archivo adjunto no se extraerá.

Coherencia de datos

Según los filtros especificados en el archivo YAML de la canalización, los proyectos (o espacios) seleccionados se extraerán una vez y se sincronizarán completamente con el receptor de destino. Luego, la supervisión continua de los cambios capturará los cambios a medida que se produzcan y actualizará los datos del receptor. Una excepción es que la supervisión de cambios sincroniza solo las acciones `create` y `update`, no las acciones `delete`.

Limitaciones

- Las acciones de eliminación de usuarios no se sincronizarán. Una vez registrados en el receptor, los datos permanecerán allí. Las actualizaciones sobrescribirán el contenido existente con nuevos cambios si se especifica la asignación de ID en la configuración del receptor.
- Las instancias en las instalaciones que utilizan versiones anteriores del software de Atlassian que no son compatibles con las siguientes API no son compatibles con este origen:
  - API de búsqueda de Jira, versión 3
    - `rest/api/3/search`
    - `rest/api/3/issue`
  - Confluence
    - `wiki/rest/api/content/search`
    - `wiki/rest/api/content`
    - `wiki/rest/api/settings/systemInfo`

Métricas en CloudWatch para conectores de Atlassian

Tipo: métricas del conector de Jira

Origen	Métrica	Tipo de métrica
acknowledgementSetSuccesses.count	Contador	Si los acuses de recibo están habilitados, esta métrica proporciona el número de tickets que se sincronizaron correctamente.

Origen	Métrica	Tipo de métrica
acknowledgementSetFailures.count	Contador	Si las confirmaciones están habilitadas, esta métrica proporciona el número de tickets que no se pudieron sincronizar.
crawlingTime.avg	Timer	El tiempo que se tardó en revisar todos los cambios nuevos.
ticketFetchLatency.avg	Timer	El promedio de latencia de la API de búsqueda de tickets.
ticketFetchLatency.max	Timer	La latencia máxima de la API de búsqueda de tickets.
ticketsRequested.count	Contador	Número de solicitudes de recogida de entradas realizadas.
ticketRequestedFailed.count	Contador	Número de solicitudes de recuperación de entradas fallidas.
ticketRequestedSuccess.count	Contador	Número de solicitudes de recuperación de entradas realizadas correctamente.
searchCallLatency.avg	Timer	Latencia media de llamadas a la API de búsqueda.
searchCallLatency.max	Timer	Latencia máxima de llamadas a la API de búsqueda.

Origen	Métrica	Tipo de métrica
searchResultsFound.count	Contador	Número de elementos encontrados en una llamada de búsqueda determinada.
searchRequestFailed.count	Contador	Los errores en las llamadas a la API de búsqueda cuentan.
authFailures.count	Contador	Número de errores de autenticación.

Tipo: métricas del conector de Confluence

Origen	Métrica	Tipo de métrica
acknowledgementSetSuccesses.count	Contador	Si las confirmaciones están habilitadas, esta métrica indica el número de páginas que se sincronizaron correctamente.
acknowledgementSetFailures.count	Contador	Si las confirmaciones están habilitadas, esta métrica proporciona el número de páginas que no se pudieron sincronizar.
crawlingTime.avg	Timer	El tiempo que se tardó en revisar todos los cambios nuevos.
pageFetchLatency.avg	Timer	Latencia de la API de obtención de contenido (media).
pageFetchLatency.max	Timer	Latencia de la API de obtención de contenido (máxima).

Origen	Métrica	Tipo de métrica
pagesRequested.count	Contador	Número de invocaciones de la API de obtención de contenido.
pageRequestFailed.count	Contador	Número de solicitudes fallidas de la API de obtención de contenido.
pageRequestedSuccess.count	Contador	Número de solicitudes satisfactorias de la API de obtención de contenido.
searchCallLatency.avg	Timer	Latencia media de llamadas a la API de búsqueda.
searchCallLatency.max	Timer	Latencia máxima de llamadas a la API de búsqueda.
searchResultsFound.count	Contador	Número de elementos encontrados en una llamada de búsqueda determinada.
searchRequestsFailed.count	Contador	Los errores en las llamadas a la API de búsqueda cuentan.
authFailures.count	Contador	Número de errores de autenticación.

Conexión de una canalización de Amazon OpenSearch Ingestion a Atlassian Jira o Confluence mediante OAuth 2.0

Utilice la información de este tema como ayuda para configurar y conectar una canalización de Amazon OpenSearch Ingestion a una cuenta de Jira o Confluence mediante la autenticación de

OAuth 2.0. Realice esta tarea cuando esté completando los the section called “Requisitos previos” para el uso de una canalización de OpenSearch Ingestion con Atlassian Services, pero opte por no usar las credenciales de la clave de la API.

Temas

- Crear una aplicación de integración de OAuth 2.0
- Generación y actualización de un token de acceso para desarrolladores de Atlassian

Crear una aplicación de integración de OAuth 2.0

Use el siguiente procedimiento para crear una aplicación de integración de OAuth 2.0 en el sitio web para desarrolladores de Atlassian.

Cómo crear una aplicación de integración de OAuth 2.0

1. Inicie sesión en su cuenta de desarrollador de Atlassian en https://developer.atlassian.com/console/myapps/.
2. Elija Crear, Integración de OAuth 2.0.
3. En Nombre, escriba un nombre para identificar el propósito de la aplicación.
4. Seleccione la casilla Acepto regirme por las condiciones del desarrollador de Atlassian y seleccione Crear.
5. En el panel de navegación de la izquierda, seleccione Autorización y, a continuación, seleccione Agregar.
6. En URL de devolución de llamada, ingrese cualquier URL, como **`https://www.amazon.com`** o **`https://www.example.com`** y seleccione Guardar cambios.
7. En el panel de navegación de la izquierda, seleccione la página Permisos y, a continuación, en la fila correspondiente a la API de Jira, seleccione Agregar. Luego, seleccione Configurar. Seleccione todos los permisos de lectura de ámbitos clásicos (la lista aparece abajo) y, a continuación, seleccione “Guardar”.
8. Seleccione la pestaña Ámbitos granulares y, a continuación, seleccione Editar ámbitos para abrir el cuadro de diálogo Editar la API de Jira.
9. Seleccione los permisos del complemento de origen que está utilizando:

   Jira

   ```
   read:audit-log:jira
   ```

```
read:issue:jira
read:issue-meta:jira
read:attachment:jira
read:comment:jira
read:comment.property:jira
read:field:jira
read:field.default-value:jira
read:field.option:jira
read:field-configuration-scheme:jira
read:field-configuration:jira
read:issue-link:jira
read:issue-link-type:jira
read:issue-link-type:jira
read:issue.remote-link:jira
read:issue.property:jira
read:resolution:jira
read:issue-details:jira
read:issue-type:jira
read:issue-worklog:jira
read:issue-field-values:jira
read:issue.changelog:jira
read:issue.transition:jira
read:issue.vote:jira
read:jira-expressions:jira
```

Confluence

```
read:content:confluence
read:content-details:confluence
read:space-details:confluence
read:audit-log:confluence
read:page:confluence
read:blogpost:confluence
read:custom-content:confluence
read:comment:confluence
read:space:confluence
read:space.property:confluence
read:space.setting:confluence
read:content.property:confluence
read:content.metadata:confluence
read:task:confluence
read:whiteboard:confluence
read:app-data:confluence
```

```
manage:confluence-configuration
```

10. Seleccione Save.

Para obtener información relacionada, consulte Implementing OAuth 2.0 (3LO) y Determining the scopes required for an operation en el sitio web para desarrolladores de Atlassian.

Generación y actualización de un token de acceso para desarrolladores de Atlassian

Utilice el siguiente procedimiento para generar y actualizar un token de acceso para desarrolladores de Atlassian en el sitio web para desarrolladores de Atlassian.

Para generar y actualizar un token de acceso de Jira

1. Inicie sesión en su cuenta de desarrollador de Atlassian en https://developer.atlassian.com/console/myapps/.
2. Seleccione la aplicación que ha creado en the section called “Crear una aplicación de integración de OAuth 2.0”.
3. En el panel de navegación de la izquierda, seleccione Autorización.
4. Copie el valor detallado de la URL de autorización de la API de Atlassian de la parte inferior de la página y péguelo en el editor de texto que prefiera.

   El formato de URL es el siguiente:

   ```
   https://auth.atlassian.com/authorize?
   audience=api.atlassian.com
   &client_id=YOUR_CLIENT_ID
   &scope=REQUESTED_SCOPE%20REQUESTED_SCOPE_TWO
   &redirect_uri=https://YOUR_APP_CALLBACK_URL
   &state=YOUR_USER_BOUND_VALUE
   &response_type=code
   &prompt=consent
   ```

5. Para `state=YOUR_USER_BOUND_VALUE`, cambie el valor del parámetro por el que prefiera, como state="**`sample_text`**".

   Para obtener más información, consulte ¿Para qué se utiliza el parámetro de estado? en el sitio web para desarrolladores de Atlassian.

6. Tenga en cuenta que en la sección `scope` se enumeran los ámbitos detallados que seleccionó en una tarea anterior. Por ejemplo: .: `scope=read%3Ajira-work%20read%3Ajira-user%20offline_access`

   `offline_access` indica que desea generar un `refresh_token`.

7. Abra una ventana del navegador Web e introduzca la URL de autorización que haya copiado en la barra de direcciones de la ventana del navegador.
8. Cuando se abra la página de destino, compruebe que la información es correcta y, a continuación, seleccione Aceptar para que lo redirijan a su página de inicio de Jira o Confluence.
9. Una vez que se haya cargado la página de inicio, copia la URL de esta página. Contiene el código de autorización de su aplicación. Utilice este código para generar su token de acceso. La sección completa siguiente a `code=` es el código de autorización.
10. Utilice el siguiente comando cURL para generar el token de acceso. Reemplace los *`valores de marcador de posición`* con su información.

    > **Tip**
    >
    > También puede usar un servicio de terceros, como Postman.

    ```
    curl --request POST --url 'https://auth.atlassian.com/oauth/token' \
    --header 'Content-Type: application/json' \
    --data '{"grant_type": "authorization_code",
    "client_id": "YOUR_CLIENT_ID",
    "client_secret": "YOUR_CLIENT_SECRET",
    "code": "AUTHORIZATION_CODE",
    "redirect_uri": "YOUR_CALLBACK_URL"}'
    ```

    La respuesta a este comando incluye los valores de `access_code` y `refresh_token`.

Uso de una canalización de OpenSearch Ingestion con Amazon Aurora

Puede utilizar una canalización de OpenSearch Ingestion con Amazon Aurora para exportar los datos existentes y transmitir los cambios (como crear, actualizar y eliminar) a los dominios y recopilaciones de Amazon OpenSearch Service. La canalización de OpenSearch Ingestion incorpora una infraestructura de captura de datos de cambios (CDC) para proporcionar una forma a gran

escala y de baja latencia de transmitir datos de forma continua desde Amazon Aurora . Aurora MySQL y Aurora PostgreSQL son compatibles.

Existen dos formas de utilizar Amazon Aurora como origen para procesar datos: con o sin una instantánea inicial completa. Una instantánea inicial completa es una instantánea de las tablas especificadas y esta instantánea se exporta a Amazon S3. Desde allí, una canalización de OpenSearch Ingestion la envía a un índice de un dominio o la particiona en varios índices en un dominio. Para mantener la coherencia de los datos en Amazon Aurora y OpenSearch, la canalización sincroniza todos los eventos de creación, actualización y eliminación de la tabla de Amazon Aurora con los documentos guardados en el índice o los índices de OpenSearch.

Cuando utiliza una instantánea inicial completa, la canalización de OpenSearch Ingestion ingiere primero la instantánea y, a continuación, comienza a leer los datos de los flujos de cambios de Amazon Aurora. Con el tiempo, recupera y mantiene la coherencia de datos casi en tiempo real entre Amazon Aurora y OpenSearch.

También puede usar la integración de OpenSearch Ingestion con Amazon Aurora para realizar un seguimiento de los cambios, capturar datos e ingerir todas las actualizaciones de Aurora en OpenSearch. Elija esta opción si ya tiene una instantánea completa de algún otro mecanismo o si solo quiere capturar todos los cambios en los datos del clúster de Amazon Aurora.

Si elige esta opción, debe configurar el registro binario para Aurora MySQL o configurar la replicación lógica para Aurora PostgreSQL en el clúster.

Temas

- Aurora MySQL
- Aurora PostgreSQL

Aurora MySQL

Complete los siguientes pasos para configurar una canalización de OpenSearch Ingestion con Amazon Aurora para Aurora MySQL.

Temas

- Requisitos previos de Aurora MySQL
- Paso 1: configurar el rol de canalización
- Paso 2: crear la canalización
- Coherencia de datos

- Asignación de tipos de datos
- Limitaciones
- Alarmas de CloudWatch recomendadas

Requisitos previos de Aurora MySQL

Antes de crear su canalización de OpenSearch Ingestion, realice los siguientes pasos:

1. Cree un grupo de parámetros de clúster de base de datos de Aurora en Amazon Aurora para configurar el registro binario.

   ```
   aurora_enhanced_binlog=1
   binlog_backup=0
   binlog_format=ROW
   binlog_replication_globaldb=0
   binlog_row_image=full
   binlog_row_metadata=full
   ```

   Además, compruebe que el parámetro `binlog_transaction_compression` no esté establecido en `ON` y que el parámetro `binlog_row_value_options` no esté establecido en `PARTIAL_JSON`.
2. Seleccione o cree un clúster de base de datos Aurora MySQL y asocie el grupo de parámetros creado en el paso anterior al clúster de base de datos.
3. Configure la retención de registros binarios en 24 horas o más.
4. Configure la autenticación con nombre de usuario y contraseña en su clúster de Amazon Aurora mediante la administración de contraseñas con Aurora y AWS Secrets Manager. También puede crear una combinación de nombre de usuario y contraseña mediante la creación de un secreto de Secrets Manager.
5. Si utiliza la característica de instantáneas iniciales completa, cree una AWS KMS key y un rol de IAM para exportar datos de Amazon Aurora a Amazon S3.

   El rol de IAM debe tener la siguiente política de permisos:

   JSON

   ```
   {
       "Version":"2012-10-17",
       "Statement": [
   ```

```
        {
            "Sid": "ExportPolicy",
            "Effect": "Allow",
            "Action": [
                "s3:PutObject*",
                "s3:ListBucket",
                "s3:GetObject*",
                "s3:DeleteObject*",
                "s3:GetBucketLocation"
            ],
            "Resource": [
                "arn:aws:s3:::s3-bucket-used-in-pipeline",
                "arn:aws:s3:::s3-bucket-used-in-pipeline/*"
            ]
        }
    ]
}
```

El rol también debe tener las siguientes relaciones de confianza:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "export.rds.amazonaws.com"
            },
            "Action": "sts:AssumeRole"
        }
    ]
}
```

6. Seleccione o cree un dominio de OpenSearch Service o una colección de OpenSearch sin servidor. Para obtener más información, consulte Creating OpenSearch Service domains y Creating collections.
7. Adjunte una política basada en recursos al dominio o una política de acceso a datos a la colección. Estas políticas de acceso permiten a OpenSearch Ingestion escribir datos de su clúster de Amazon Aurora en su dominio o colección.

Paso 1: configurar el rol de canalización

Una vez configurados los requisitos previos de la canalización de Amazon Aurora, configure el rol de la canalización que se usará en la configuración de la canalización. Agregue también los siguientes permisos para el origen de Amazon Aurora al rol:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
    {
    "Sid": "allowReadingFromS3Buckets",
    "Effect": "Allow",
    "Action": [
    "s3:GetObject",
    "s3:DeleteObject",
    "s3:GetBucketLocation",
    "s3:ListBucket",
    "s3:PutObject"
    ],
    "Resource": [
    "arn:aws:s3:::s3_bucket",
    "arn:aws:s3:::s3_bucket/*"
    ]
    },
    {
    "Sid": "allowNetworkInterfacesActions",
    "Effect": "Allow",
    "Action": [
    "ec2:AttachNetworkInterface",
    "ec2:CreateNetworkInterface",
    "ec2:CreateNetworkInterfacePermission",
    "ec2:DeleteNetworkInterface",
    "ec2:DeleteNetworkInterfacePermission",
    "ec2:DetachNetworkInterface",
    "ec2:DescribeNetworkInterfaces"
    ],
    "Resource": [
    "arn:aws:ec2:*:111122223333:network-interface/*",
    "arn:aws:ec2:*:111122223333:subnet/*",
    "arn:aws:ec2:*:111122223333:security-group/*"
    ]
```

```
    },
    {
    "Sid": "allowDescribeEC2",
    "Effect": "Allow",
    "Action": [
    "ec2:Describe*"
    ],
    "Resource": "*"
    },
    {
    "Sid": "allowTagCreation",
    "Effect": "Allow",
    "Action": [
    "ec2:CreateTags"
    ],
    "Resource": "arn:aws:ec2:*:111122223333:network-interface/*",
    "Condition": {
    "StringEquals": {
    "aws:RequestTag/OSISManaged": "true"
    }
    }
    },
    {
    "Sid": "AllowDescribeInstances",
    "Effect": "Allow",
    "Action": [
    "rds:DescribeDBInstances"
    ],
    "Resource": [
    "arn:aws:rds:us-east-2:111122223333:db:*"
    ]
    },
    {
    "Sid": "AllowDescribeClusters",
    "Effect": "Allow",
    "Action": [
    "rds:DescribeDBClusters"
    ],
    "Resource": [
    "arn:aws:rds:us-east-2:111122223333:cluster:DB-id"
    ]
    },
    {
    "Sid": "AllowSnapshots",
```

```
"Effect": "Allow",
"Action": [
"rds:DescribeDBClusterSnapshots",
"rds:CreateDBClusterSnapshot",
"rds:AddTagsToResource"
],
"Resource": [
"arn:aws:rds:us-east-2:111122223333:cluster:DB-id",
"arn:aws:rds:us-east-2:111122223333:cluster-snapshot:DB-id*"
]
},
{
"Sid": "AllowExport",
"Effect": "Allow",
"Action": [
"rds:StartExportTask"
],
"Resource": [
"arn:aws:rds:us-east-2:111122223333:cluster:DB-id",
"arn:aws:rds:us-east-2:111122223333:cluster-snapshot:DB-id*"
]
},
{
"Sid": "AllowDescribeExports",
"Effect": "Allow",
"Action": [
"rds:DescribeExportTasks"
],
"Resource": "*",
"Condition": {
"StringEquals": {
"aws:RequestedRegion": "us-east-2",
"aws:ResourceAccount": "111122223333"
}
}
},
{
"Sid": "AllowAccessToKmsForExport",
"Effect": "Allow",
"Action": [
"kms:Decrypt",
"kms:Encrypt",
"kms:DescribeKey",
"kms:RetireGrant",
```

```
    "kms:CreateGrant",
    "kms:ReEncrypt*",
    "kms:GenerateDataKey*"
    ],
    "Resource": [
    "arn:aws:kms:us-east-2:111122223333:key/export-key-id"
    ]
    },
    {
    "Sid": "AllowPassingExportRole",
    "Effect": "Allow",
    "Action": "iam:PassRole",
    "Resource": [
    "arn:aws:iam::111122223333:role/export-role"
    ]
    },
    {
    "Sid": "SecretsManagerReadAccess",
    "Effect": "Allow",
    "Action": [
    "secretsmanager:GetSecretValue"
    ],
    "Resource": [
    "arn:aws:secretsmanager:*:111122223333:secret:*"
    ]
    }
    ]
    }
```

Paso 2: crear la canalización

Configure una canalización de OpenSearch Ingestion similar a la siguiente. La canalización de ejemplo especifica un clúster de Amazon Aurora como origen.

```
version: "2"
aurora-mysql-pipeline:
  source:
    rds:
      db_identifier: "cluster-id"
      engine: aurora-mysql
      database: "database-name"
      tables:
```

```
        include:
          - "table1"
          - "table2"
      s3_bucket: "bucket-name"
      s3_region: "bucket-region"
      s3_prefix: "prefix-name"
      export:
        kms_key_id: "kms-key-id"
        iam_role_arn: "export-role-arn"
      stream: true
      aws:
        sts_role_arn: "arn:aws:iam::account-id:role/pipeline-role"
        region: "us-east-1"
      authentication:
        username: ${{aws_secrets:secret:username}}
        password: ${{aws_secrets:secret:password}}
  sink:
    - opensearch:
        hosts: ["https://search-mydomain.us-east-1.es.amazonaws.com"]
        index: "${getMetadata(\"table_name\")}"
        index_type: custom
        document_id: "${getMetadata(\"primary_key\")}"
        action: "${getMetadata(\"opensearch_action\")}"
        document_version: "${getMetadata(\"document_version\")}"
        document_version_type: "external"
        aws:
          sts_role_arn: "arn:aws:iam::account-id:role/pipeline-role"
          region: "us-east-1"
extension:
  aws:
    secrets:
      secret:
        secret_id: "rds-secret-id"
        region: "us-east-1"
        sts_role_arn: "arn:aws:iam::account-id:role/pipeline-role"
        refresh_interval: PT1H
```

Puede utilizar un esquema de Amazon Aurora preconfigurado para crear esta canalización. Para obtener más información, consulte Uso de esquemas.

A fin de utilizar Amazon Aurora como origen, debe configurar el acceso a la VPC para la canalización. La VPC que elija debe ser la misma que usa su origen de Amazon Aurora. A continuación, elija una o varias subredes y uno o varios grupos de seguridad de VPC. Tenga en

cuenta que la canalización necesita acceso de red a una base de datos Aurora MySQL, por lo que también debe comprobar que el clúster de Aurora esté configurado con un grupo de seguridad de VPC que permita el tráfico entrante desde el grupo de seguridad de VPC de la canalización al puerto de la base de datos. Para obtener más información, consulte Control de acceso con grupos de seguridad.

Si utiliza la Consola de administración de AWS para crear la canalización, también debe adjuntar la canalización a la VPC para poder usar Amazon Aurora como origen. Para ello, busque la sección Configuración de la red, seleccione la casilla Adjuntar a la VPC y elija el CIDR de una de las opciones predeterminadas proporcionadas o seleccione una opción propia. Puede usar cualquier CIDR de un espacio de direcciones privado, tal como se define en RFC 1918 Best Current Practice.

Para proporcionar un CIDR personalizado, seleccione Otros en el menú desplegable. Para evitar una colisión de direcciones IP entre OpenSearch Ingestion y Amazon Aurora, asegúrese de que el CIDR de la VPC de Amazon Aurora sea diferente del CIDR de OpenSearch Ingestion.

Para obtener más información, consulte Configuring VPC access for a pipeline.

Coherencia de datos

La canalización garantiza la coherencia de datos mediante el sondeo continuo o la recepción de cambios del clúster de Amazon Aurora y la actualización de los documentos correspondientes en el índice de OpenSearch.

OpenSearch Ingestion admite el reconocimiento integral para garantizar la durabilidad de los datos. Cuando una canalización lee instantáneas o transmisiones, crea particiones de forma dinámica para el procesamiento paralelo. La canalización marca una partición como completa cuando recibe un reconocimiento después de incorporar todos los registros en el dominio o la colección de OpenSearch. Si desea incorporarlos a una colección de búsquedas de OpenSearch sin servidor, puede generar un ID de documento en la canalización. Si quiere ingerirlos en una colección de series temporales de OpenSearch sin servidor, tenga en cuenta que la canalización no genera un ID de documento, de modo que debe omitir `document_id: "${getMetadata(\"primary_key\")}"` en la configuración del receptor de la canalización.

Una canalización de OpenSearch Ingestion también asigna las acciones de los eventos entrantes a las correspondientes acciones de indexación masiva para facilitar la incorporación de documentos. Esto mantiene la coherencia de datos, de modo que cada cambio de datos en Amazon Aurora se concilia con los cambios de documento correspondientes en OpenSearch.

Asignación de tipos de datos

La canalización de OpenSearch Ingestion asigna los tipos de datos de MySQL a representaciones que son adecuadas para el consumo de los dominios o las colecciones de OpenSearch Service. Si no se ha definido ninguna plantilla de asignación en OpenSearch, OpenSearch determina automáticamente los tipos de campo con una asignación dinámica en función del primer documento enviado. También puede definir de forma explícita los tipos de campo que mejor se adapten a sus necesidades en OpenSearch mediante una plantilla de asignación.

En la siguiente tabla, se muestran los tipos de datos de MySQL y los tipos de campos de OpenSearch correspondientes. La columna Tipo de campo de OpenSearch predeterminado muestra el tipo de campo correspondiente en OpenSearch si no se ha definido una asignación explícita. En este caso, OpenSearch determina automáticamente los tipos de campo mediante una asignación dinámica. La columna Tipo de campo de OpenSearch recomendado es el tipo de campo correspondiente que se recomienda especificar de forma explícita en una plantilla de asignación. Estos tipos de campos están más alineados con los tipos de datos de MySQL y, por lo general, pueden habilitar mejores características de búsqueda disponibles en OpenSearch.

Tipo de datos de MySQL	Tipo de campo de OpenSearch predeterminado	Tipo de campo de OpenSearch recomendado
BIGINT	long	long
BIGINT UNSIGNED	long	largo sin signo
BIT	long	byte, corto, entero o largo según el número de bits
DECIMAL	text	doble o palabra
DOUBLE	float	double
FLOAT	float	float
INT	long	entero

Tipo de datos de MySQL	Tipo de campo de OpenSearch predeterminado	Tipo de campo de OpenSearch recomendado
INT UNSIGNED	long	long
MEDIUMIN	long	entero
MEDIUMIN UNSIGNED	long	entero
NUMERIC	text	doble o palabra
SMALLINT	long	short
SMALLINT UNSIGNED	long	entero
TINYINT	long	byte
TINYINT UNSIGNED	long	short
BINARIO	text	binario
BLOB	text	binario
CHAR	text	text
ENUM	text	palabra clave
LONGBLOE	text	binario
LONGTEXT	text	text
MEDIUMBL B	text	binario
MEDIUMTE T	text	text

Tipo de datos de MySQL	Tipo de campo de OpenSearch predeterminado	Tipo de campo de OpenSearch recomendado
SET	text	palabra clave
TEXT	text	text
TINYBLOB	text	binario
TINYTEXT	text	text
VARBINAR	text	binario
VARCHAR	text	text
DATE	largo (en milisegundos de época)	date
DATETIME	largo (en milisegundos de época)	date
TIME	largo (en milisegundos de época)	date
TIMESTAM	largo (en milisegundos de época)	date
YEAR	largo (en milisegundos de época)	date
GEOMETR	texto (en formato WKT)	geo_shape
GEOMETR OLLECTIO	texto (en formato WKT)	geo_shape
LINESTRIN G	texto (en formato WKT)	geo_shape
MULTILINE STRING	texto (en formato WKT)	geo_shape
MULTIPOI T	texto (en formato WKT)	geo_shape

Tipo de datos de MySQL	Tipo de campo de OpenSearch predeterminado	Tipo de campo de OpenSearch recomendado
MULTIPOLYGON	texto (en formato WKT)	geo_shape
POINT	texto (en formato WKT)	geo_point o geo_shape
POLYGON	texto (en formato WKT)	geo_shape
JSON	text	objeto

Le recomendamos configurar la cola de mensajes fallidos (DLQ) en su canalización de OpenSearch Ingestion. Si ha configurado la cola, OpenSearch Service envía todos los documentos con errores que no se pueden incorporar debido a errores de asignación dinámica.

Si las asignaciones automáticas fallan, puede usar `template_type` y `template_content` en su configuración de canalización para definir reglas de asignación explícitas. Como alternativa, puede crear plantillas de asignación directamente en su dominio o colección de búsqueda antes de iniciar la canalización.

Limitaciones

Tenga en cuenta las siguientes limitaciones cuando configure una canalización de OpenSearch Ingestion para Aurora MySQL:

- La integración solo admite una base de datos MySQL por canalización.
- La integración no admite actualmente la ingesta de datos entre regiones; su clúster de Amazon Aurora y dominio de OpenSearch deben estar en la misma Región de AWS.
- La integración no admite actualmente la ingesta de datos entre cuentas; su clúster de Amazon Aurora y canalización de OpenSearch Ingestion deben estar en la misma Cuenta de AWS.
- Asegúrese de que el clúster de Amazon Aurora tenga habilitada la autenticación mediante Secrets Manager, que es el único mecanismo de autenticación admitido.
- La configuración de canalización existente no se puede actualizar para ingerir datos de una base de datos diferente o de una tabla diferente. Para actualizar la base de datos o el nombre de la tabla de una canalización, debe detener la canalización y reiniciarla con una configuración actualizada o crear una canalización nueva.

- Por lo general, las instrucciones en lenguaje de definición de datos (DDL) no son compatibles. La coherencia de datos no se mantendrá si:
  - Se cambian las claves principales (agregar/eliminar/cambiar el nombre).
  - Las tablas se eliminan o se truncan.
  - Se cambian los nombres de las columnas o los tipos de datos.
- Si las tablas MySQL que se van a sincronizar no tienen definidas las claves principales, no se garantiza la coherencia de datos. Deberá definir correctamente la opción `document_id` personalizada en la configuración del receptor de OpenSearch para poder sincronizar las actualizaciones o eliminaciones con OpenSearch.
- No se admiten las referencias a claves externas con acciones de eliminación en cascada y pueden provocar inconsistencias en los datos entre Aurora MySQL y OpenSearch.
- Versiones compatibles: Aurora MySQL 3.05.2 y versiones posteriores.

Alarmas de CloudWatch recomendadas

Se recomiendan las siguientes métricas de CloudWatch para supervisar el rendimiento de la canalización de ingesta. Estas métricas pueden ayudarlo a identificar la cantidad de datos procesados a partir de exportaciones y flujos, los errores al procesar eventos de transmisión y exportación y el número de documentos escritos en el destino. Puede configurar las alarmas de CloudWatch para que lleven a cabo una acción cuando una de estas métricas supere un valor especificado durante un periodo de tiempo determinado.

Métrica	Descripción
pipeline-name .rds.credentialsChanged	Esta métrica indica la frecuencia con la que se rotan los secretos de AWS.
pipeline-name .rds.executorRefreshErrors	Esta métrica indica errores al actualizar los secretos de AWS.
pipeline-name .rds.exportRecordsTotal	Esta métrica indica el número de registros exportados desde Amazon Aurora.

Métrica	Descripción
pipeline-name .rds.exportRecordsProcessed	Esta métrica indica el número de registros procesados por la canalización de OpenSearch Ingestion.
pipeline-name .rds.exportRecordProcessingErrors	Esta métrica indica el número de errores de procesamiento en una canalización de OpenSearch Ingestion al leer los datos de un clúster de Amazon Aurora.
pipeline-name .rds.exportRecordsSuccessTotal	Esta métrica indica el número total de registros de exportación procesados correctamente.
pipeline-name .rds.exportRecordsFailedTotal	Esta métrica indica el número total de registros que no se han podido procesar.
pipeline-name .rds.bytesReceived	Esta métrica indica el número total de bytes recibidos por una canalización de OpenSearch Ingestion.
pipeline-name .rds.bytesProcessed	Esta métrica indica el número total de bytes procesados por una canalización de OpenSearch Ingestion.
pipeline-name .rds.streamRecordsSuccessTotal	Esta métrica indica el número de registros del flujo procesados correctamente.
pipeline-name .rds.streamRecordsFailedTotal	Esta métrica indica el número total de registros del flujo que no se han podido procesar.

Aurora PostgreSQL

Complete los siguientes pasos para configurar una canalización de OpenSearch Ingestion con Amazon Aurora para Aurora PostgreSQL.

Temas

- Requisitos previos de Aurora PostgreSQL
- Paso 1: configurar el rol de canalización
- Paso 2: crear la canalización
- Coherencia de datos
- Asignación de tipos de datos
- Limitaciones
- Alarmas de CloudWatch recomendadas

Requisitos previos de Aurora PostgreSQL

Antes de crear su canalización de OpenSearch Ingestion, realice los siguientes pasos:

1. Cree un grupo de parámetros de clúster de base de datos personalizado en Amazon Aurora para configurar la replicación lógica.

   ```
   rds.logical_replication=1
       aurora.enhanced_logical_replication=1
       aurora.logical_replication_backup=0
       aurora.logical_replication_globaldb=0
   ```

2. Seleccione o cree un clúster de base de datos Aurora PostgreSQL y asocie el grupo de parámetros creado en el paso 1 al clúster de base de datos.
3. Configure la autenticación con nombre de usuario y contraseña en su clúster de Amazon Aurora mediante la administración de contraseñas con Aurora y AWS Secrets Manager. También puede crear una combinación de nombre de usuario y contraseña mediante la creación de un secreto de Secrets Manager.
4. Si utiliza la característica de instantáneas iniciales completa, cree una AWS KMS key y un rol de IAM para exportar datos de Amazon Aurora a Amazon S3.

   El rol de IAM debe tener la siguiente política de permisos:

JSON

```
{
        "Version":"2012-10-17",
        "Statement": [
            {
                "Sid": "ExportPolicy",
                "Effect": "Allow",
                "Action": [
                    "s3:PutObject*",
                    "s3:ListBucket",
                    "s3:GetObject*",
                    "s3:DeleteObject*",
                    "s3:GetBucketLocation"
                ],
                "Resource": [
                    "arn:aws:s3:::s3-bucket-used-in-pipeline",
                    "arn:aws:s3:::s3-bucket-used-in-pipeline/*"
                ]
            }
        ]
    }
```

El rol también debe tener las siguientes relaciones de confianza:

JSON

```
{
        "Version":"2012-10-17",
        "Statement": [
            {
                "Effect": "Allow",
                "Principal": {
                    "Service": "export.rds.amazonaws.com"
                },
                "Action": "sts:AssumeRole"
            }
        ]
    }
```

5. Seleccione o cree un dominio de OpenSearch Service o una colección de OpenSearch sin servidor. Para obtener más información, consulte Creating OpenSearch Service domains y Creating collections.
6. Adjunte una política basada en recursos al dominio o una política de acceso a datos a la colección. Estas políticas de acceso permiten a OpenSearch Ingestion escribir datos de su clúster de Amazon Aurora en su dominio o colección.

Paso 1: configurar el rol de canalización

Una vez configurados los requisitos previos de la canalización de Amazon Aurora, configure el rol de la canalización que se usará en la configuración de la canalización. Agregue también los siguientes permisos para el origen de Amazon Aurora al rol:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
    {
    "Sid": "allowReadingFromS3Buckets",
    "Effect": "Allow",
    "Action": [
    "s3:GetObject",
    "s3:DeleteObject",
    "s3:GetBucketLocation",
    "s3:ListBucket",
    "s3:PutObject"
    ],
    "Resource": [
    "arn:aws:s3:::s3_bucket",
    "arn:aws:s3:::s3_bucket/*"
    ]
    },
    {
    "Sid": "allowNetworkInterfacesActions",
    "Effect": "Allow",
    "Action": [
    "ec2:AttachNetworkInterface",
    "ec2:CreateNetworkInterface",
    "ec2:CreateNetworkInterfacePermission",
```

```
    "ec2:DeleteNetworkInterface",
    "ec2:DeleteNetworkInterfacePermission",
    "ec2:DetachNetworkInterface",
    "ec2:DescribeNetworkInterfaces"
    ],
    "Resource": [
    "arn:aws:ec2:*:111122223333:network-interface/*",
    "arn:aws:ec2:*:111122223333:subnet/*",
    "arn:aws:ec2:*:111122223333:security-group/*"
    ]
    },
    {
    "Sid": "allowDescribeEC2",
    "Effect": "Allow",
    "Action": [
    "ec2:Describe*"
    ],
    "Resource": "*"
    },
    {
    "Sid": "allowTagCreation",
    "Effect": "Allow",
    "Action": [
    "ec2:CreateTags"
    ],
    "Resource": "arn:aws:ec2:*:111122223333:network-interface/*",
    "Condition": {
    "StringEquals": {
    "aws:RequestTag/OSISManaged": "true"
    }
    }
    },
    {
    "Sid": "AllowDescribeInstances",
    "Effect": "Allow",
    "Action": [
    "rds:DescribeDBInstances"
    ],
    "Resource": [
    "arn:aws:rds:us-east-2:111122223333:db:*"
    ]
    },
    {
    "Sid": "AllowDescribeClusters",
```

```
"Effect": "Allow",
"Action": [
"rds:DescribeDBClusters"
],
"Resource": [
"arn:aws:rds:us-east-2:111122223333:cluster:DB-id"
]
},
{
"Sid": "AllowSnapshots",
"Effect": "Allow",
"Action": [
"rds:DescribeDBClusterSnapshots",
"rds:CreateDBClusterSnapshot",
"rds:AddTagsToResource"
],
"Resource": [
"arn:aws:rds:us-east-2:111122223333:cluster:DB-id",
"arn:aws:rds:us-east-2:111122223333:cluster-snapshot:DB-id*"
]
},
{
"Sid": "AllowExport",
"Effect": "Allow",
"Action": [
"rds:StartExportTask"
],
"Resource": [
"arn:aws:rds:us-east-2:111122223333:cluster:DB-id",
"arn:aws:rds:us-east-2:111122223333:cluster-snapshot:DB-id*"
]
},
{
"Sid": "AllowDescribeExports",
"Effect": "Allow",
"Action": [
"rds:DescribeExportTasks"
],
"Resource": "*",
"Condition": {
"StringEquals": {
"aws:RequestedRegion": "us-east-2",
"aws:ResourceAccount": "111122223333"
}
```

```
    }
    },
    {
    "Sid": "AllowAccessToKmsForExport",
    "Effect": "Allow",
    "Action": [
    "kms:Decrypt",
    "kms:Encrypt",
    "kms:DescribeKey",
    "kms:RetireGrant",
    "kms:CreateGrant",
    "kms:ReEncrypt*",
    "kms:GenerateDataKey*"
    ],
    "Resource": [
    "arn:aws:kms:us-east-2:111122223333:key/export-key-id"
    ]
    },
    {
    "Sid": "AllowPassingExportRole",
    "Effect": "Allow",
    "Action": "iam:PassRole",
    "Resource": [
    "arn:aws:iam::111122223333:role/export-role"
    ]
    },
    {
    "Sid": "SecretsManagerReadAccess",
    "Effect": "Allow",
    "Action": [
    "secretsmanager:GetSecretValue"
    ],
    "Resource": [
    "arn:aws:secretsmanager:*:111122223333:secret:*"
    ]
    }
    ]
    }
```

Paso 2: crear la canalización

Configure una canalización de OpenSearch Ingestion como la siguiente, que especifica un clúster de Aurora PostgreSQL como origen.

```
version: "2"
aurora-postgres-pipeline:
  source:
    rds:
      db_identifier: "cluster-id"
      engine: aurora-postgresql
      database: "database-name"
      tables:
        include:
          - "schema1.table1"
          - "schema2.table2"
      s3_bucket: "bucket-name"
      s3_region: "bucket-region"
      s3_prefix: "prefix-name"
      export:
        kms_key_id: "kms-key-id"
        iam_role_arn: "export-role-arn"
      stream: true
      aws:
        sts_role_arn: "arn:aws:iam::account-id:role/pipeline-role"
        region: "us-east-1"
      authentication:
        username: ${{aws_secrets:secret:username}}
        password: ${{aws_secrets:secret:password}}
  sink:
    - opensearch:
        hosts: ["https://search-mydomain.us-east-1.es.amazonaws.com"]
        index: "${getMetadata(\"table_name\")}"
        index_type: custom
        document_id: "${getMetadata(\"primary_key\")}"
        action: "${getMetadata(\"opensearch_action\")}"
        document_version: "${getMetadata(\"document_version\")}"
        document_version_type: "external"
        aws:
          sts_role_arn: "arn:aws:iam::account-id:role/pipeline-role"
          region: "us-east-1"
extension:
  aws:
    secrets:
```

```
      secret:
        secret_id: "rds-secret-id"
        region: "us-east-1"
        sts_role_arn: "arn:aws:iam::account-id:role/pipeline-role"
        refresh_interval: PT1H
```

> **Note**
>
> Puede utilizar un esquema de Amazon Aurora preconfigurado para crear esta canalización. Para obtener más información, consulte Uso de esquemas.

A fin de utilizar Amazon Aurora como origen, debe configurar el acceso a la VPC para la canalización. La VPC que elija debe ser la misma que usa su origen de Amazon Aurora. A continuación, elija una o varias subredes y uno o varios grupos de seguridad de VPC. Tenga en cuenta que la canalización necesita acceso de red a una base de datos Aurora MySQL, por lo que también debe comprobar que el clúster de Aurora esté configurado con un grupo de seguridad de VPC que permita el tráfico entrante desde el grupo de seguridad de VPC de la canalización al puerto de la base de datos. Para obtener más información, consulte Control de acceso con grupos de seguridad.

Si utiliza la Consola de administración de AWS para crear la canalización, también debe adjuntar la canalización a la VPC para poder usar Amazon Aurora como origen. Para ello, busque la sección Configuración de la red, seleccione Adjuntar a la VPC y elija el CIDR de una de las opciones predeterminadas proporcionadas o seleccione una opción propia. Puede usar cualquier CIDR de un espacio de direcciones privado, tal como se define en RFC 1918 Best Current Practice.

Para proporcionar un CIDR personalizado, seleccione Otros en el menú desplegable. Para evitar una colisión de direcciones IP entre OpenSearch Ingestion y Amazon Aurora, asegúrese de que el CIDR de la VPC de Amazon Aurora sea diferente del CIDR de OpenSearch Ingestion.

Para obtener más información, consulte Configuring VPC access for a pipeline.

Coherencia de datos

La canalización garantiza la coherencia de datos mediante el sondeo continuo o la recepción de cambios del clúster de Amazon Aurora y la actualización de los documentos correspondientes en el índice de OpenSearch.

OpenSearch Ingestion admite el reconocimiento integral para garantizar la durabilidad de los datos. Cuando una canalización lee instantáneas o transmisiones, crea particiones de forma dinámica para el procesamiento paralelo. La canalización marca una partición como completa cuando recibe un reconocimiento después de incorporar todos los registros en el dominio o la colección de OpenSearch. Si desea incorporarlos a una colección de búsquedas de OpenSearch sin servidor, puede generar un ID de documento en la canalización. Si quiere ingerirlos en una colección de series temporales de OpenSearch sin servidor, tenga en cuenta que la canalización no genera un ID de documento, de modo que debe omitir `document_id: "${getMetadata(\"primary_key\")}"` en la configuración del receptor de la canalización.

Una canalización de OpenSearch Ingestion también asigna las acciones de los eventos entrantes a las correspondientes acciones de indexación masiva para facilitar la incorporación de documentos. Esto mantiene la coherencia de datos, de modo que cada cambio de datos en Amazon Aurora se concilia con los cambios de documento correspondientes en OpenSearch.

Asignación de tipos de datos

OpenSearch Ingestion Pipeline asigna los tipos de datos de Aurora PostgreSQL a representaciones que son adecuadas para el consumo de los dominios o colecciones de OpenSearch Service. Si no se ha definido ninguna plantilla de asignación en OpenSearch, OpenSearch determina automáticamente los tipos de campo con una asignación dinámica en función del primer documento enviado. También puede definir de forma explícita los tipos de campo que mejor se adapten a sus necesidades en OpenSearch mediante una plantilla de asignación.

La siguiente tabla muestra los tipos de datos de Aurora PostgreSQL y los tipos de campo de OpenSearch correspondientes. La columna Tipo de campo de OpenSearch predeterminado muestra el tipo de campo correspondiente en OpenSearch si no se ha definido una asignación explícita. En este caso, OpenSearch determina automáticamente los tipos de campo mediante una asignación dinámica. La columna Tipo de campo recomendado de OpenSearch es el tipo de campo que se recomienda especificar de forma explícita en una plantilla de asignación. Estos tipos de campos están más alineados con los tipos de datos de Aurora PostgreSQL y, por lo general, pueden habilitar mejores características de búsqueda disponibles en OpenSearch.

Tipo de datos de Aurora PostgreSQL	Tipo de campo de OpenSearch predeterminado	Tipo de campo de OpenSearch recomendado
smallint	long	short
entero	long	entero
bigint	long	long
decimal	text	doble o palabra
numeric[(p, s)]	text	doble o palabra
real	float	float
double precision	float	double
smallserial	long	short
serial	long	entero
bigserial	long	long
money	objeto	objeto
character varying(n)	text	text
varchar(n)	text	text
character (n)	text	text
char(n)	text	text

Tipo de datos de Aurora PostgreSQL	Tipo de campo de OpenSearch predeterminado	Tipo de campo de OpenSearch recomendado
bpchar (n)	text	text
BPCHAR	text	text
text	text	text
enum	text	text
bytea	text	binario
timestamp [(p)] [sin zona horaria]	largo (en milisegundos de época)	date
timestamp [(p)] con zona horaria	largo (en milisegundos de época)	date
date	largo (en milisegundos de época)	date
time [(p)] [sin zona horaria]	largo (en milisegundos de época)	date
time [(p)] con zona horaria	largo (en milisegundos de época)	date
intervalo [campos] [(p)]	texto (formato ISO8601)	text

Tipo de datos de Aurora PostgreSQL	Tipo de campo de OpenSearch predeterminado	Tipo de campo de OpenSearch recomendado
booleano	booleano	booleano
point	texto (en formato WKT)	geo_shape
línea	texto (en formato WKT)	geo_shape
LSEG	texto (en formato WKT)	geo_shape
BOX	texto (en formato WKT)	geo_shape
path	texto (en formato WKT)	geo_shape
polígono	texto (en formato WKT)	geo_shape
CIRCLE	objeto	objeto
cidr	text	text
inet	text	text
macaddr	text	text
macaddr8	text	text
bit(n)	long	byte, corto, entero o largo (según el número de bits)
bit varying(n)	long	byte, corto, entero o largo (según el número de bits)
json	objeto	objeto
jsonb	objeto	objeto
jsonpath	text	text

Le recomendamos configurar la cola de mensajes fallidos (DLQ) en su canalización de OpenSearch Ingestion. Si ha configurado la cola, OpenSearch Service envía todos los documentos con errores que no se pueden incorporar debido a errores de asignación dinámica.

En caso de que las asignaciones automáticos fallen, puede usar `template_type` y `template_content` en su configuración de canalización para definir reglas de asignación explícitas. Como alternativa, puede crear plantillas de asignación directamente en su dominio o colección de búsqueda antes de iniciar la canalización.

Limitaciones

Tenga en cuenta las siguientes limitaciones cuando configure una canalización de OpenSearch Ingestion para Aurora PostgreSQL:

- La integración solo admite una base de datos PostgreSQL de Aurora por canalización.
- La integración no admite actualmente la ingesta de datos entre regiones; su clúster de Amazon Aurora y dominio de OpenSearch deben estar en la misma Región de AWS.
- La integración no admite actualmente la ingesta de datos entre cuentas; su clúster de Amazon Aurora y canalización de OpenSearch Ingestion deben estar en la misma Cuenta de AWS.
- Asegúrese de que el clúster de Amazon Aurora tenga habilitada la autenticación mediante AWS Secrets Manager, que es el único mecanismo de autenticación admitido.
- La configuración de canalización existente no se puede actualizar para ingerir datos de una base de datos diferente o de una tabla diferente. Para actualizar la base de datos o el nombre de la tabla de una canalización, debe detener la canalización y reiniciarla con una configuración actualizada o crear una canalización nueva.
- Por lo general, las instrucciones en lenguaje de definición de datos (DDL) no son compatibles. La coherencia de datos no se mantendrá si:
  - Se cambian las claves principales (agregar/eliminar/cambiar el nombre).
  - Las tablas se eliminan o se truncan.
  - Se cambian los nombres de las columnas o los tipos de datos.
- Si las tablas de PostgreSQL de Aurora que se van a sincronizar no tienen definidas las claves principales, no se garantiza la coherencia de datos. Deberá definir la opción `document_id` personalizada en OpenSearch y asimilar correctamente la configuración para poder sincronizar las actualizaciones y eliminaciones con OpenSearch.
- Versiones compatibles: Aurora PostgreSQL 16.4 y versiones posteriores.

Alarmas de CloudWatch recomendadas

Se recomiendan las siguientes métricas de CloudWatch para supervisar el rendimiento de la canalización de ingesta. Estas métricas pueden ayudarle a identificar la cantidad de datos procesados a partir de exportaciones y flujos, los errores al procesar eventos de transmisión y exportación y el número de documentos escritos en el destino. Puede configurar las alarmas de CloudWatch para que lleven a cabo una acción cuando una de estas métricas supere un valor especificado durante un periodo de tiempo determinado.

Métrica	Descripción
pipeline-name .rds.credentialsChanged	Esta métrica indica la frecuencia con la que se rotan los secretos de AWS.
pipeline-name .rds.executorRefreshErrors	Esta métrica indica errores al actualizar los secretos de AWS.
pipeline-name .rds.exportRecordsTotal	Esta métrica indica el número de registros exportados desde Amazon Aurora.
pipeline-name .rds.exportRecordsProcessed	Esta métrica indica el número de registros procesados por la canalización de OpenSearch Ingestion.
pipeline-name .rds.exportRecordProcessingErrors	Esta métrica indica el número de errores de procesamiento en una canalización de OpenSearch Ingestion al leer los datos de un clúster de Amazon Aurora.
pipeline-name .rds.exportRecordsSuccessTotal	Esta métrica indica el número total de registros de exportación procesados correctamente.
pipeline-name .rds.expo	Esta métrica indica el número total de registros que no se han podido procesar.

Métrica	Descripción
rtRecordsFailedTotal	
pipeline-name .rds.bytesReceived	Esta métrica indica el número total de bytes recibidos por una canalización de OpenSearch Ingestion.
pipeline-name .rds.bytesProcessed	Esta métrica indica el número total de bytes procesados por una canalización de OpenSearch Ingestion.
pipeline-name .rds.streamRecordsSuccessTotal	Esta métrica indica el número de registros del flujo procesados correctamente.
pipeline-name .rds.streamRecordsFailedTotal	Esta métrica indica el número total de registros del flujo que no se han podido procesar.

Uso de una canalización de OpenSearch Ingestion con Amazon DynamoDB

Puede usar el complemento DynamoDB para transmitir eventos de tablas, como creaciones, actualizaciones y eliminaciones, a los dominios de Amazon OpenSearch Service y a las colecciones de Amazon OpenSearch sin servidor. La canalización utiliza la captura de datos de cambios (CDC) para la transmisión de alta escala y baja latencia.

Puede procesar datos de DynamoDB con o sin una instantánea inicial completa.

- Con una instantánea completa, DynamoDB utiliza la recuperación en un momento dado (PITR) para crear una copia de seguridad y la carga en Amazon S3. A continuación, OpenSearch Ingestion indexa la instantánea en uno o varios índices de OpenSearch. Para mantener la coherencia, la canalización sincroniza todos los cambios de DynamoDB con OpenSearch. Esta opción requiere que habilite PITR y DynamoDB Streams.

- Sin una instantánea: OpenSearch Ingestion solo transmite los eventos nuevos de DynamoDB. Elija esta opción si ya tiene una instantánea o si necesita una transmisión en tiempo real sin datos históricos. Esta opción requiere que habilite únicamente DynamoDB Streams.

Para obtener más información, consulte Integración sin ETL de DynamoDB con Amazon OpenSearch Service en la Guía para desarrolladores de Amazon DynamoDB.

Temas

- Requisitos previos
- Paso 1: configurar el rol de canalización
- Paso 2: crear la canalización
- Coherencia de datos
- Asignación de tipos de datos
- Limitaciones
- Alarmas de CloudWatch recomendadas para DynamoDB

Requisitos previos

Para configurar la canalización, debe tener una tabla de DynamoDB con DynamoDB Streams activado. La transmisión debe usar el tipo de vista de transmisión `NEW_IMAGE`. Sin embargo, las canalizaciones de OpenSearch Ingestion también pueden transmitir eventos con `NEW_AND_OLD_IMAGES` si este tipo de vista de transmisión se ajusta a su caso de uso.

Si utiliza instantáneas, también debe habilitar la recuperación en un momento dado en su tabla. Para obtener más información, consulte Creación de una tabla, Habilitación de la recuperación en un momento dado y Habilitación de una transmisión en la Guía para desarrolladores de Amazon DynamoDB.

Paso 1: configurar el rol de canalización

Una vez configurada la tabla de DynamoDB, configure el rol de canalización que desee usar en la configuración de canalización y añada los siguientes permisos de DynamoDB al rol:

JSON

```
{
```

```
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "allowRunExportJob",
            "Effect": "Allow",
            "Action": [
                "dynamodb:DescribeTable",
                "dynamodb:DescribeContinuousBackups",
                "dynamodb:ExportTableToPointInTime"
            ],
            "Resource": [
                "arn:aws:dynamodb:us-east-1:111122223333:table/my-table"
            ]
        },
        {
            "Sid": "allowCheckExportjob",
            "Effect": "Allow",
            "Action": [
                "dynamodb:DescribeExport"
            ],
            "Resource": [
                "arn:aws:dynamodb:us-east-1:111122223333:table/my-table/export/*"
            ]
        },
        {
            "Sid": "allowReadFromStream",
            "Effect": "Allow",
            "Action": [
                "dynamodb:DescribeStream",
                "dynamodb:GetRecords",
                "dynamodb:GetShardIterator"
            ],
            "Resource": [
                "arn:aws:dynamodb:us-east-1:111122223333:table/my-table/stream/*"
            ]
        },
        {
            "Sid": "allowReadAndWriteToS3ForExport",
            "Effect": "Allow",
            "Action": [
                "s3:GetObject",
                "s3:AbortMultipartUpload",
                "s3:PutObject",
                "s3:PutObjectAcl"
```

```
            ],
            "Resource": [
                "arn:aws:s3:us-east-1:111122223333:amzn-s3-demo-bucket/export-
folder/*"
            ]
        }
    ]
}
```

También puede usar una clave administrada por el cliente de AWS KMS para cifrar los archivos de datos de exportación. Para descifrar los objetos exportados, especifique `s3_sse_kms_key_id` para el ID de clave en la configuración de exportación de la canalización con el siguiente formato: `arn:aws:kms:region:account-id:key/my-key-id`. La siguiente política incluye los permisos necesarios para utilizar una clave administrada por el cliente:

```
{
    "Sid": "allowUseOfCustomManagedKey",
    "Effect": "Allow",
    "Action": [
        "kms:GenerateDataKey",
        "kms:Decrypt"
    ],
    "Resource": arn:aws:kms:region:account-id:key/my-key-id
}
```

Paso 2: crear la canalización

A continuación, puede configurar una canalización de OpenSearch Ingestion como la siguiente, en la que se especifica a DynamoDB como el origen. Este ejemplo de canalización incorpora los datos de la `table-a` con la instantánea de PITR, seguidos de los eventos de DynamoDB Streams. Una posición inicial de `LATEST` indica que la canalización debe leer los datos más recientes de DynamoDB Streams.

```
version: "2"
cdc-pipeline:
  source:
    dynamodb:
      tables:
      - table_arn: "arn:aws:dynamodb:region:account-id:table/table-a"
        export:
```

```
          s3_bucket: "my-bucket"
          s3_prefix: "export/"
        stream:
          start_position: "LATEST"
      aws:
        region: "us-east-1"
  sink:
  - opensearch:
      hosts: ["https://search-mydomain.region.es.amazonaws.com"]
      index: "${getMetadata(\"table-name\")}"
      index_type: custom
      normalize_index: true
      document_id: "${getMetadata(\"primary_key\")}"
      action: "${getMetadata(\"opensearch_action\")}"
      document_version: "${getMetadata(\"document_version\")}"
      document_version_type: "external"
```

Puede utilizar un esquema preconfigurado de DynamoDB para crear esta canalización. Para obtener más información, consulte the section called “Uso de esquemas”.

Coherencia de datos

OpenSearch Ingestion admite el reconocimiento integral para garantizar la durabilidad de los datos. Cuando una canalización lee instantáneas o transmisiones, crea particiones de forma dinámica para el procesamiento paralelo. La canalización marca una partición como completa cuando recibe un reconocimiento después de incorporar todos los registros en el dominio o la colección de OpenSearch.

Si desea incorporarlos a una colección de búsquedas de OpenSearch sin servidor, puede generar un ID de documento en la canalización. Si quiere incorporarlos a una colección de series temporales de OpenSearch sin servidor, tenga en cuenta que la canalización no genera un ID de documento.

Una canalización de OpenSearch Ingestion también asigna las acciones de los eventos entrantes a las correspondientes acciones de indexación masiva para facilitar la incorporación de documentos. Esto mantiene la coherencia de datos, de modo que cada cambio de datos en DynamoDB se concilie con los cambios de documento correspondientes en OpenSearch.

Asignación de tipos de datos

OpenSearch Service asigna de manera dinámica los tipos de datos de cada documento entrante al tipo de datos correspondiente en DynamoDB. En la siguiente tabla se muestra cómo OpenSearch Service asigna de forma automática varios tipos de datos.

Tipo de datos	OpenSearch	DynamoDB
Número	OpenSearch asigna automáticamente los datos numéricos. Si el número es un número entero, OpenSearch lo asigna como un valor largo. Si el número es fraccionario, OpenSearch lo asigna como un valor flotante. OpenSearch asigna de manera dinámica varios atributos en función del primer documento enviado. Si tiene una combinación de tipos de datos para el mismo atributo en DynamoDB, como un número entero y un número fraccionario, es posible que se produzca un error en la asignación. Por ejemplo, si el primer documento tiene un atributo que es un número entero y un documento posterior tiene el mismo atributo como un número fraccionario, OpenSearch no podrá incorporar el segundo documento. En estos casos, debe proporcionar una plantilla de asignación explícita, como la siguiente: `{ "template": { "mappings": { "properties": { "MixedNumberAttribute": { "type": "float" } } } }`	DynamoDB admite números.

Tipo de datos	OpenSearch	DynamoDB
	<pre> } }</pre> Si necesita una precisión doble, utilice la asignación de campos de tipo cadena. No hay ningún tipo numérico equivalente que admita 38 dígitos de precisión en OpenSearch.	
Conjunto de números	OpenSearch asigna automáticamente un conjunto de números a una matriz de valores largos o flotantes. Al igual que con los números escalares, esto depende de si el primer número incorporado es un número entero o un número fraccionario. Puede proporcionar asignaciones para conjuntos de números de la misma manera que asigna cadenas escalares.	DynamoDB admite tipos que representan conjuntos de números.

Tipo de datos	OpenSearch	DynamoDB
Cadena	OpenSearch asigna automáticamente los valores de cadena en forma de texto. En algunas situaciones, como en el caso de los valores enumerados, puede asignarlos al tipo de palabra clave. En el siguiente ejemplo se muestra cómo asignar un atributo de DynamoDB denominado `PartType` a una palabra clave de OpenSearch. `{ "template": { "mappings": { "properties": { "PartType": { "type": "keyword" } } } } }`	DynamoDB admite cadenas.
Conjunto de cadenas	OpenSearch asigna de manera automática un conjunto de cadenas en una matriz de cadenas. Puede proporcionar asignaciones para conjuntos de cadenas de la misma manera que asigna cadenas escalares.	DynamoDB admite tipos que representan conjuntos de cadenas.

Tipo de datos	OpenSearch	DynamoDB
Binario	OpenSearch asigna de manera automática los datos binarios en forma de texto. Puede proporcionar una asignación para escribirlos como campos binarios en OpenSearch. En el siguiente ejemplo se muestra cómo asignar un atributo de DynamoDB denominado `ImageData` a un campo binario de OpenSearch. `{ "template": { "mappings": { "properties": { "ImageData": { "type": "binary" } } } } }`	DynamoDB es compatible con los atributos de tipo binarios.
Conjunto binario	OpenSearch asigna de manera automática un conjunto binario a una matriz de datos binarios en forma de texto. Puede proporcionar asignaciones para conjuntos de números de la misma manera que asigna el binario escalar.	DynamoDB admite tipos que represent an conjuntos de valores binarios.
Booleano	OpenSearch asigna un tipo booleano de DynamoDB a un tipo booleano de OpenSearch.	DynamoDB admite atributos de tipo booleano.

Tipo de datos	OpenSearch	DynamoDB
Nulo	OpenSearch puede incorporar documentos del tipo nulos de DynamoDB. Guarda el valor como un valor nulo en el documento. No hay ninguna asignación para este tipo y este campo no está indexado ni se puede buscar en él. Si se utiliza el mismo nombre de atributo para un tipo nulo y, posteriormente, se cambia a un tipo diferente, como una cadena, OpenSearch crea una asignación dinámica para el primer valor no nulo. Los valores subsiguientes pueden seguir siendo valores nulos de DynamoDB.	DynamoDB admite atributos de tipo nulo.

Tipo de datos	OpenSearch	DynamoDB
Asignación	OpenSearch asigna los atributos de asignación de DynamoDB a campos anidados. Las mismas asignaciones se aplican dentro de un campo anidado. En el siguiente ejemplo se asigna una cadena de un campo anidado a un tipo de palabra clave en OpenSearch: `{ "template": { "mappings": { "properties": { "AdditionalDescriptions": { "properties": { "PartType": { "type": "keyword" } } } } } } }`	DynamoDB admite atributos de tipo asignación.

Tipo de datos	OpenSearch	DynamoDB
Enumeración	OpenSearch proporciona resultados diferentes para las listas de DynamoDB, según el contenido de la lista. Cuando una lista contiene todos los tipos escalares del mismo tipo (por ejemplo, una lista de todas las cadenas), OpenSearch incorpora la lista como una matriz de ese tipo. Esto funciona para los tipos cadena, número, booleano y nulo. Las restricciones para cada uno de estos tipos son las mismas que las restricciones para un escalar de ese tipo. También puede proporcionar asignaciones para listas de mapas con la misma asignación que usaría para un mapa. No puede proporcionar una lista de tipos mixtos.	DynamoDB admite atributos de tipo lista.

Tipo de datos	OpenSearch	DynamoDB
Establezca	OpenSearch proporciona resultados diferentes para los conjuntos de DynamoDB en función del contenido del conjunto. Cuando un conjunto contiene todos los tipos escalares del mismo tipo (por ejemplo, un conjunto de todas las cadenas), entonces OpenSearch incorpora el conjunto como una matriz de ese tipo. Esto funciona para los tipos cadena, número, booleano y nulo. Las restricciones para cada uno de estos tipos son las mismas que las restricciones para un escalar de ese tipo. También puede proporcionar asignaciones para conjuntos de mapas con la misma asignación que usaría para un mapa. No puede proporcionar un conjunto de tipos mixtos.	DynamoDB admite tipos que representan conjuntos.

Le recomendamos configurar la cola de mensajes fallidos (DLQ) en su canalización de OpenSearch Ingestion. Si ha configurado la cola, OpenSearch Service envía todos los documentos con errores que no se pueden incorporar debido a errores de asignación dinámica.

En caso de que las asignaciones automáticos fallen, puede usar `template_type` y `template_content` en su configuración de canalización para definir reglas de asignación explícitas. Como alternativa, puede crear plantillas de asignación directamente en su dominio o colección de búsqueda antes de iniciar la canalización.

Limitaciones

Tenga en cuenta las siguientes limitaciones cuando configure una canalización de OpenSearch Ingestion para DynamoDB:

- En la actualidad, la integración de OpenSearch Ingestion con DynamoDB no admite la ingesta entre regiones. La tabla de DynamoDB y la canalización de OpenSearch Ingestion deben estar en la misma Región de AWS.
- La tabla de DynamoDB y la canalización de OpenSearch Ingestion deben estar en la misma Cuenta de AWS.
- Una canalización de OpenSearch Ingestion solo admite una tabla de DynamoDB como su origen.
- DynamoDB Streams solo almacena datos en un registro durante un máximo de 24 horas. Si la ingesta de una instantánea inicial de una tabla grande demora 24 horas o más, se producirá una pérdida inicial de datos. Para mitigar esta pérdida de datos, calcule el tamaño de la tabla y configure las unidades de computación adecuadas de las canalizaciones de OpenSearch Ingestion.

Alarmas de CloudWatch recomendadas para DynamoDB

Se recomiendan las siguientes métricas de CloudWatch para supervisar el rendimiento de la canalización de ingesta. Estas métricas pueden ayudarle a identificar la cantidad de datos procesados a partir de exportaciones y flujos, los errores al procesar eventos de transmisión y exportación y la cantidad de documentos escritos en el destino. Puede configurar las alarmas de CloudWatch para que lleven a cabo una acción cuando una de estas métricas supere un valor especificado durante un periodo de tiempo determinado.

Métrica	Descripción
`dynamodb-pipeline.BlockingBuffer.bufferUsage.value`	Indica qué cantidad del búfer se está utilizando.
`dynamodb-pipeline.dynamodb.activeExportS3ObjectConsumers.value`	Muestra el número total de OCU que procesan activamente objetos de Amazon S3 para la exportación.
`dynamodb-pipeline.dynamodb.bytesProcessed.count`	Recuento de bytes procesados desde el origen de DynamoDB.

Métrica	Descripción
dynamodb-pipeline.dynamodb.changeEventsProcessed.count	Número de eventos de cambio procesados desde el flujo de DynamoDB.
dynamodb-pipeline.dynamodb.changeEventsProcessingErrors.count	Número de errores de eventos de cambio procesados desde DynamoDB.
dynamodb-pipeline.dynamodb.exportJobFailure.count	Number of export job submission attempts that have failed.
dynamodb-pipeline.dynamodb.exportJobSuccess.count	Number of export jobs that have been submitted successfully.
dynamodb-pipeline.dynamodb.exportRecordsProcessed.count	Número total de registros procesados desde la exportación.
dynamodb-pipeline.dynamodb.exportRecordsTotal.count	Número total de registros exportados desde DynamoDB, esencial para el seguimiento de los volúmenes de exportación de datos.
dynamodb-pipeline.dynamodb.exportS3ObjectsProcessed.count	Total number of export data files that have been processed successfully from Amazon S3.
dynamodb-pipeline.opensearch.bulkBadRequestErrors.count	Count of errors during bulk requests due to malformed request.
dynamodb-pipeline.opensearch.bulkRequestLatency.avg	Average latency for bulk write requests made to OpenSearch.
dynamodb-pipeline.opensearch.bulkRequestNotFoundErrors.count	Number of bulk requests that failed because the target data could not be found.
dynamodb-pipeline.opensearch.bulkRequestNumberOfRetries.count	Number of retries by OpenSearch Ingestion pipelines to write OpenSearch cluster.

Métrica	Descripción
dynamodb-pipeline.opensearch.bulkRequestSizeBytes.sum	Total size in bytes of all bulk requests made to OpenSearch.
dynamodb-pipeline.opensearch.documentErrors.count	Number of errors when sending documents to OpenSearch. The documents causing the errors witll be sent to DLQ.
dynamodb-pipeline.opensearch.documentsSuccess.count	Number of documents successfully written to an OpenSearch cluster or collection.
dynamodb-pipeline.opensearch.documentsSuccessFirstAttempt.count	Number of documents successfully indexed in OpenSearch on the first attempt.
dynamodb-pipeline.opensearch.documentsVersionConflictErrors.count	Count of errors due to version conflicts in documents during processing.
dynamodb-pipeline.opensearch.PipelineLatency.avg	Average latency of OpenSearch Ingestion pipeline to process the data by reading from the source to writint to the destination.
dynamodb-pipeline.opensearch.PipelineLatency.max	Maximum latency of OpenSearch Ingestion pipeline to process the data by reading from the source to writing the destination.
dynamodb-pipeline.opensearch.recordsIn.count	Count of records successfully ingested into OpenSearch. This metric is essential for tracking the volume of data being processed and stored.
dynamodb-pipeline.opensearch.s3.dlqS3RecordsFailed.count	Number of records that failed to write to DLQ.
dynamodb-pipeline.opensearch.s3.dlqS3RecordsSuccess.count	Number of records that are written to DLQ.

Métrica	Descripción
`dynamodb-pipeline.opensearch.s3.dlqS3RequestLatency.count`	Count of latency measurements for requests to the Amazon S3 dead-letter queue.
`dynamodb-pipeline.opensearch.s3.dlqS3RequestLatency.sum`	Total latency for all requests to the Amazon S3 dead-letter queue
`dynamodb-pipeline.opensearch.s3.dlqS3RequestSizeBytes.sum`	Total size in bytes of all requests made to the Amazon S3 dead-letter queue.
`dynamodb-pipeline.recordsProcessed.count`	Total number of records processed in the pipeline, a key metric for overal throughput.
`dynamodb.changeEventsProcessed.count`	No records are being gathered from DynamoDB streams. This could be due to no activitiy on the table, an export being in progress, or an issue accessing the DynamoDB streams.
`dynamodb.exportJobFailure.count`	The attempt to trigger an export to S3 failed.
`dynamodb-pipeline.opensearch.bulkRequestInvalidInputErrors.count`	Count of bulk request errors in OpenSearch due to invalid input, crucial for monitoring data quality and operational issues.
`opensearch.EndToEndLatency.avg`	The end to end latnecy is higher than desired for reading from DynamoDB streams. This could be due to an underscaled OpenSearch cluster or a maximum pipeline OCU capacity that is too low for the WCU throughput on the DynamoDB table. This end to end latency will be high after an export and should decrease over time as it catches up to the latest DynamoDB streams.

Uso de una canalización de OpenSearch Ingestion con Amazon DocumentDB

Puede usar el complemento DocumentDB para transmitir los cambios en los documentos, como las creaciones, actualizaciones y eliminaciones, a Amazon OpenSearch Service. La canalización admite la captura de datos de cambios (CDC), si está disponible, o el sondeo de API para la transmisión a alta escala y baja latencia.

Puede procesar los datos con o sin una instantánea inicial completa. Una instantánea completa captura una colección completa de Amazon DocumentDB y la carga en Amazon S3. A continuación, la canalización envía los datos a uno o varios índices de OpenSearch. Una vez que ingiere la instantánea, la canalización sincroniza los cambios en curso para mantener la coherencia y, finalmente, recupera las actualizaciones casi en tiempo real.

Si ya tiene una instantánea completa de otro origen o solo necesita procesar eventos nuevos, puede transmitir sin una instantánea. En este caso, la canalización lee directamente los flujos de cambios de Amazon DocumentDB sin una carga masiva inicial.

Debe habilitar una transmisión de cambios en la recopilación de Amazon DocumentDB si habilita una transmisión. Sin embargo, si solo realiza una carga completa o una exportación, no necesitará ningún flujo de cambios.

Requisitos previos

Antes de crear su canalización de OpenSearch Ingestion, realice los siguientes pasos:

1. Para crear un clúster de Amazon DocumentDB con permiso para leer datos, siga los pasos que se indican en Creación de un clúster de Amazon DocumentDB en la Guía para desarrolladores de Amazon DocumentDB. Si utiliza la infraestructura de CDC, configure su clúster de Amazon DocumentDB para publicar flujos de cambios.
2. Habilitar TLS en su clúster de Amazon DocumentDB.
3. Configure un CIDR de VPC de un espacio de direcciones privado para usarlo con OpenSearch Ingestion.
4. Configure la autenticación en el clúster de Amazon DocumentDB con AWS Secrets Manager. Para habilitar la rotación de secretos, siga los pasos de Automatically Rotating Passwords for Amazon DocumentDB. Para obtener más información, consulte Acceso a la base de datos mediante el control de acceso basado en roles y Seguridad en Amazon DocumentDB.

5. Si utiliza un flujo de cambios para suscribirse a los cambios de datos de la colección de Amazon DocumentDB, puede ampliar el periodo de retención hasta 7 días con el parámetro `change_stream_log_retention_duration` para evitar la pérdida de datos. Los eventos de flujos de cambios se almacenan durante 3 horas, de forma predeterminada, una vez registrado el evento, lo que no es tiempo suficiente para colecciones de gran tamaño. Para modificar el periodo de retención del flujo de cambios, consulte Modifying the change stream log retention duration.
6. Cree un dominio de OpenSearch Service o una colección de OpenSearch sin servidor. Para obtener más información, consulte the section called " Creación de dominios de OpenSearch Service" y the section called "Creación de colecciones".
7. Adjunte una política basada en recursos al dominio o una política de acceso a datos a la colección. Estas políticas de acceso permiten a OpenSearch Ingestion escribir datos de su clúster de Amazon DocumentDB en su dominio o colección.

   El siguiente ejemplo de política de acceso al dominio permite que el rol de canalización, que creará en el paso siguiente, escriba datos en un dominio. Asegúrese de actualizar el `resource` con su propio ARN.

   JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::444455556666:role/pipeline-role"
      },
      "Action": [
        "es:DescribeDomain",
        "es:ESHttp*"
      ],
      "Resource": [
        "arn:aws:es:us-east-1:111122223333:domain/domain-name"
      ]
    }
  ]
}
```

   Para crear un rol de IAM con los permisos correctos para acceder a los datos de escritura de la colección o dominio, consulte the section called "Configuración de roles y usuarios".

Paso 1: configurar el rol de canalización

Una vez configurados los requisitos previos de la canalización de Amazon DocumentDB, configure el rol de la canalización que desee usar en la configuración de la canalización y agregue los siguientes permisos de Amazon DocumentDB al rol:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "allowS3ListObjectAccess",
            "Effect": "Allow",
            "Action": [
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::s3-bucket"
            ],
            "Condition": {
                "StringLike": {
                    "s3:prefix": "s3-prefix/*"
                }
            }
        },
        {
            "Sid": "allowReadAndWriteToS3ForExportStream",
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:GetObject",
                "s3:DeleteObject"
            ],
            "Resource": [
                "arn:aws:s3:::s3-bucket/s3-prefix/*"
            ]
        },
        {
            "Sid": "SecretsManagerReadAccess",
            "Effect": "Allow",
            "Action": [
```

```
                "secretsmanager:GetSecretValue"
            ],
            "Resource": [
                "arn:aws:secretsmanager:us-east-1:111122223333:secret:secret-
name"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:AttachNetworkInterface",
                "ec2:CreateNetworkInterface",
                "ec2:CreateNetworkInterfacePermission",
                "ec2:DeleteNetworkInterface",
                "ec2:DeleteNetworkInterfacePermission",
                "ec2:DetachNetworkInterface",
                "ec2:DescribeNetworkInterfaces"
            ],
            "Resource": [
                "arn:aws:ec2:*:111122223333:network-interface/*",
                "arn:aws:ec2:*:111122223333:subnet/*",
                "arn:aws:ec2:*:111122223333:security-group/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeDhcpOptions",
                "ec2:DescribeRouteTables",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeSubnets",
                "ec2:DescribeVpcs",
                "ec2:Describe*"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateTags"
            ],
            "Resource": "arn:aws:ec2:*:*:network-interface/*",
            "Condition": {
                "StringEquals": {
```

```
                    "aws:RequestTag/OSISManaged": "true"
                }
            }
        }
    ]
}
```

Debe proporcionar los permisos de Amazon EC2 anteriores en el rol de IAM que utilice para crear la canalización de OpenSearch Ingestion, ya que la canalización utiliza estos permisos para crear y eliminar una interfaz de red en la VPC. La canalización solo puede acceder al clúster de Amazon DocumentDB a través de esta interfaz de red.

Paso 2: crear la canalización

A continuación, puede configurar una canalización de OpenSearch Ingestion como la siguiente, en la que se especifica Amazon DocumentDB como origen. Tenga en cuenta que, para rellenar el nombre del índice, la función `getMetadata` utiliza *`documentdb_collection`* como clave de metadatos. Si desea usar un nombre de índice diferente sin el método `getMetadata`, puede usar la configuración `index: "`*`my_index_name`*`"`.

```
version: "2"
documentdb-pipeline:
  source:
    documentdb:
      acknowledgments: true
      host: "https://docdb-cluster-id.us-east-1.docdb.amazonaws.com"
      port: 27017
      authentication:
        username: ${aws_secrets:secret:username}
        password: ${aws_secrets:secret:password}
      aws:
      s3_bucket: "bucket-name"
      s3_region: "bucket-region"
      s3_prefix: "path" #optional path for storing the temporary data
      collections:
        - collection: "dbname.collection"
          export: true
          stream: true
  sink:
  - opensearch:
      hosts: ["https://search-mydomain.us-east-1.es.amazonaws.com"]
```

```
      index: "${getMetadata(\"documentdb_collection\")}"
      index_type: custom
      document_id: "${getMetadata(\"primary_key\")}"
      action: "${getMetadata(\"opensearch_action\")}"
      document_version: "${getMetadata(\"document_version\")}"
      document_version_type: "external"
extension:
  aws:
    secrets:
      secret:
        secret_id: "my-docdb-secret"
        region: "us-east-1"
        refresh_interval: PT1H
```

Puede utilizar un esquema de Amazon DocumentDB preconfigurado para crear esta canalización. Para obtener más información, consulte the section called “Uso de esquemas”.

Si utiliza la Consola de administración de AWS para crear la canalización, también debe adjuntar la canalización a la VPC para poder usar Kafka Amazon DocumentDB como origen. Para ello, busque la sección Opciones de la red de origen, seleccione la casilla Attach to VPC (Adjuntar a la VPC) y elija el CIDR de una de las opciones predeterminadas proporcionadas. Puede usar cualquier CIDR de un espacio de direcciones privado, tal como se define en RFC 1918 Best Current Practice.

Para proporcionar un CIDR personalizado, seleccione Otros en el menú desplegable. Para evitar una colisión de direcciones IP entre OpenSearch Ingestion y Amazon DocumentDB, asegúrese de que el CIDR de la VPC de Amazon DocumentDB sea diferente del CIDR de OpenSearch Ingestion.

Para obtener más información, consulte Configuring VPC access for a pipeline.

Coherencia de datos

La canalización garantiza la coherencia de datos mediante el sondeo continuo o la recepción de cambios del clúster de Amazon DocumentDB y la actualización de los documentos correspondientes en el índice de OpenSearch.

OpenSearch Ingestion admite el reconocimiento integral para garantizar la durabilidad de los datos. Cuando una canalización lee instantáneas o transmisiones, crea particiones de forma dinámica para el procesamiento paralelo. La canalización marca una partición como completa cuando recibe un reconocimiento después de incorporar todos los registros en el dominio o la colección de OpenSearch.

Si desea incorporarlos a una colección de búsquedas de OpenSearch sin servidor, puede generar un ID de documento en la canalización. Si quiere ingerirlos en una colección de series temporales de OpenSearch sin servidor, tenga en cuenta que la canalización no genera un ID de documento, de modo que debe omitir `document_id: "${getMetadata(\"primary_key\")}"` en la configuración del receptor de la canalización.

Una canalización de OpenSearch Ingestion también asigna las acciones de los eventos entrantes a las correspondientes acciones de indexación masiva para facilitar la incorporación de documentos. Esto mantiene la coherencia de datos, de modo que cada cambio de datos en Amazon DocumentDB se concilia con los cambios de documento correspondientes en OpenSearch.

Asignación de tipos de datos

OpenSearch Service asigna de manera dinámica los tipos de datos de cada documento entrante al tipo de datos correspondiente en Amazon DocumentDB. En la siguiente tabla se muestra cómo OpenSearch Service asigna de forma automática varios tipos de datos.

Tipo de datos	OpenSearch	Amazon DocumentDB
Entero	OpenSearch asigna automátic amente los valores de enteros de Amazon DocumentDB a los enteros de OpenSearch. OpenSearch asigna de manera dinámica el campo en función del primer documento enviado. Si tiene una combinación de tipos de datos para el mismo atributo en Amazon DocumentD B, es posible que se produzca un error en la asignación automática. Por ejemplo, si el primer documento tiene un atributo que es un largo y un documento posterior tiene el mismo atributo como un entero, OpenSearch no podrá ingerir el segundo documento . En estos casos, debe proporcionar una	Amazon DocumentDB admite enteros.

Tipo de datos	OpenSearch	Amazon DocumentDB
	plantilla de asignación explícita que elija el tipo de número más flexible, como la siguiente: `{ "template": { "mappings": { "properties": { "MixedNumberField": { "type": "float" } } } } }`	

Tipo de datos	OpenSearch	Amazon DocumentDB
Largo	OpenSearch asigna automáticamente los valores largos de Amazon DocumentDB a los largos de OpenSearch. OpenSearch asigna de manera dinámica el campo en función del primer documento enviado. Si tiene una combinación de tipos de datos para el mismo atributo en Amazon DocumentDB, es posible que se produzca un error en la asignación automática. Por ejemplo, si el primer documento tiene un atributo que es un largo y un documento posterior tiene el mismo atributo como un entero, OpenSearch no podrá ingerir el segundo documento. En estos casos, debe proporcionar una plantilla de asignación explícita que elija el tipo de número más flexible, como la siguiente: `{ "template": { "mappings": { "properties": { "MixedNumberField": { "type": "float" } } } } }`	Amazon DocumentDB admite largos.

Tipo de datos	OpenSearch	Amazon DocumentDB
Cadena	OpenSearch asigna automáticamente los valores de cadena en forma de texto. En algunas situaciones, como en el caso de los valores enumerados, puede asignarlos al tipo de palabra clave. En el siguiente ejemplo se muestra cómo asignar un atributo de Amazon DocumentDB denominado `PartType` a una palabra clave de OpenSearch. `{ "template": { "mappings": { "properties": { "PartType": { "type": "keyword" } } } } }`	Amazon DocumentDB admite cadenas.

Tipo de datos	OpenSearch	Amazon DocumentDB
Double	OpenSearch asigna automáticamente los valores dobles de Amazon DocumentDB a los dobles de OpenSearch. OpenSearch asigna de manera dinámica el campo en función del primer documento enviado. Si tiene una combinación de tipos de datos para el mismo atributo en Amazon DocumentDB, es posible que se produzca un error en la asignación automática. Por ejemplo, si el primer documento tiene un atributo que es un largo y un documento posterior tiene el mismo atributo como un entero, OpenSearch no podrá ingerir el segundo documento. En estos casos, debe proporcionar una plantilla de asignación explícita que elija el tipo de número más flexible, como la siguiente: `{ "template": { "mappings": { "properties": { "MixedNumberField": { "type": "float" } } } } }`	Amazon DocumentDB admite dobles.

Tipo de datos	OpenSearch	Amazon DocumentDB
Date	De forma predeterminada, la fecha se asigna a un entero en OpenSearch. Puede definir una plantilla de asignación personalizada para asignar una fecha a una fecha de OpenSearch. `{ "template": { "mappings": { "properties": { "myDateField": { "type": "date", "format": "epoch_second" } } } } }`	Amazon DocumentDB admite fechas.

Tipo de datos	OpenSearch	Amazon DocumentDB
Marca de tiempo	De forma predeterminada, la marca temporal se asigna a un entero en OpenSearch. Puede definir una plantilla de asignación personalizada para asignar una fecha a una fecha de OpenSearch. `{ "template": { "mappings": { "properties": { "myTimestampField": { "type": "date", "format": "epoch_second" } } } } }`	Amazon DocumentDB admite marcas temporales.
Booleano	OpenSearch asigna un tipo booleano de Amazon DocumentDB a un tipo booleano de OpenSearch.	Amazon DocumentDB admite atributos de tipo booleano.

Tipo de datos	OpenSearch	Amazon DocumentDB
Decimal	OpenSearch asigna los atributos de asignación de Amazon DocumentDB a campos anidados. Las mismas asignaciones se aplican dentro de un campo anidado. En el siguiente ejemplo se asigna una cadena de un campo anidado a un tipo de palabra clave en OpenSearch: `{ "template": { "mappings": { "properties": { "myDecimalField": { "type": "double" } } } } }` Con esta asignación personalizada, puede consultar y agregar el campo con una precisión de doble nivel. El valor original retiene toda la precisión de la propiedad `_source` del documento de OpenSearch. Sin esta asignación, OpenSearch usa el texto de forma predeterminada.	Amazon DocumentDB admite decimales.
Expresión regular	El tipo de expresión regular crea campos anidados. Entre ellos se encuentran *<myFieldName>* `.pattern` y *<myFieldName>* `.options`.	Amazon DocumentDB admite expresiones regulares.

Tipo de datos	OpenSearch	Amazon DocumentDB
Datos Binary	OpenSearch asigna automáticamente los datos binarios de Amazon DocumentDB a texto de OpenSearch. Puede proporcionar una asignación para escribirlos como campos binarios en OpenSearch. En el siguiente ejemplo se muestra cómo asignar un campo de Amazon DocumentDB denominado `imageData` a un campo binario de OpenSearch. `{ "template": { "mappings": { "properties": { "imageData": { "type": "binary" } } } } }`	Amazon DocumentDB admite campos de datos binarios.
ObjectId	Los campos con un tipo de objectId se asignan a los campos de texto de OpenSearch. El valor será la representación en cadena del objectId.	Amazon DocumentDB admite objectId.

Tipo de datos	OpenSearch	Amazon DocumentDB
Nulo	OpenSearch puede ingerir documentos de tipo nulo de Amazon DocumentDB. Guarda el valor como un valor nulo en el documento. No hay ninguna asignación para este tipo y este campo no está indexado ni se puede buscar en él. Si se utiliza el mismo nombre de atributo para un tipo nulo y, posteriormente, se cambia a un tipo diferente, como una cadena, OpenSearch crea una asignación dinámica para el primer valor no nulo. Los valores subsiguientes pueden seguir siendo valores nulos de Amazon DocumentDB.	Amazon DocumentDB admite campos de tipo nulo.
Sin definir	OpenSearch puede ingerir documentos de tipo indefinido de Amazon DocumentDB. Guarda el valor como un valor nulo en el documento. No hay ninguna asignación para este tipo y este campo no está indexado ni se puede buscar en él. Si se utiliza el mismo nombre de campo para un tipo indefinido y, posteriormente, se cambia a un tipo diferente, como una cadena, OpenSearch crea una asignación dinámica para el primer valor que no esté sin definir. Los valores subsiguientes pueden seguir siendo valores indefinidos de Amazon DocumentDB.	Amazon DocumentDB admite campos de tipo indefinido.

Tipo de datos	OpenSearch	Amazon DocumentDB
MinKey	OpenSearch puede ingerir documentos de tipo minKey de Amazon DocumentDB. Guarda el valor como un valor nulo en el documento. No hay ninguna asignación para este tipo y este campo no está indexado ni se puede buscar en él. Si se utiliza el mismo nombre de campo para un tipo minKey y, posteriormente, se cambia a un tipo diferente, como una cadena, OpenSearch crea una asignación dinámica para el primer valor no minKey. Los valores subsiguientes pueden seguir siendo valores minKey de Amazon DocumentDB.	Amazon DocumentDB admite campos de tipo minKey.
MaxKey	OpenSearch puede ingerir documentos de tipo maxKey de Amazon DocumentDB. Guarda el valor como un valor nulo en el documento. No hay ninguna asignación para este tipo y este campo no está indexado ni se puede buscar en él. Si se utiliza el mismo nombre de campo para un tipo maxKey y, posteriormente, se cambia a un tipo diferente, como una cadena, OpenSearch crea una asignación dinámica para el primer valor no maxKey. Los valores subsiguientes pueden seguir siendo valores maxKey de Amazon DocumentDB.	Amazon DocumentDB admite campos de tipo maxKey.

Le recomendamos configurar la cola de mensajes fallidos (DLQ) en su canalización de OpenSearch Ingestion. Si ha configurado la cola, OpenSearch Service envía todos los documentos con errores que no se pueden incorporar debido a errores de asignación dinámica.

En caso de que las asignaciones automáticos fallen, puede usar `template_type` y `template_content` en su configuración de canalización para definir reglas de asignación explícitas. Como alternativa, puede crear plantillas de asignación directamente en su dominio o colección de búsqueda antes de iniciar la canalización.

Limitaciones

Tenga en cuenta las siguientes limitaciones cuando configure una canalización de OpenSearch Ingestion para Amazon DocumentDB:

- Actualmente, la integración de OpenSearch Ingestion con Amazon DocumentDB no admite la ingesta entre regiones. Su clúster de Amazon DocumentDB y canalización de OpenSearch Ingestion deben estar en la misma Región de AWS.
- Actualmente, la integración de OpenSearch Ingestion con Amazon DocumentDB no admite la ingesta entre cuentas. Su clúster de Amazon DocumentDB y canalización de OpenSearch Ingestion deben estar en la misma Cuenta de AWS.
- Una canalización de OpenSearch Ingestion solo admite un clúster de Amazon DocumentDB como su origen.
- La integración de OpenSearch Ingestion con Amazon DocumentDB es compatible específicamente con los clústeres basados en instancias de Amazon DocumentDB. No es compatible con los clústeres elásticos de Amazon DocumentDB.
- La integración de OpenSearch Ingestion solo admite AWS Secrets Manager como mecanismo de autenticación para su clúster de Amazon DocumentDB.
- No puede actualizar la configuración de la canalización existente para ingerir datos de una base de datos o colección diferente. En su lugar, debe crear una canalización nueva.

Alarmas de CloudWatch recomendadas

Para lograr un mejor rendimiento, le recomendamos que use las siguientes alarmas de CloudWatch al crear una canalización de OpenSearch Ingestion para acceder a un clúster de Amazon DocumentDB como origen.

Alarma de CloudWatch	Descripción
<nombre-canalización>.document db.credentialsChanged	Esta métrica indica la frecuencia con la que se rotan los secretos de AWS.
<nombre-canalización>.document db.executorRefreshErrors	Esta métrica indica errores al actualizar los secretos de AWS.
<nombre-canalización>.document db.exportRecordsTotal	Esta métrica indica el número de registros exportados desde Amazon DocumentDB.
<nombre-canalización>.document db.exportRecordsProcessed	Esta métrica indica el número de registros procesados por la canalización de OpenSearch Ingestion.
<nombre-canalización>.document db.exportRecordProcessingErrors	Esta métrica indica el número de errores de procesamiento en una canalización de OpenSearch Ingestion al leer los datos de un clúster de Amazon DocumentDB.
<nombre-canalización>.document db.exportRecordsSuccessTotal	Esta métrica indica el número total de registros de exportación procesados correctamente.
<nombre-canalización>.document db.exportRecordsFailedTotal	Esta métrica indica el número total de registros que no se han podido procesar.
<nombre-canalización>.document db.bytesReceived	Esta métrica indica el número total de bytes recibidos por una canalización de OpenSearch Ingestion.
<nombre-canalización>.document db.bytesProcessed	Esta métrica indica el número total de bytes procesados por una canalización de OpenSearch Ingestion.
<nombre-canalización>.document db.exportPartitionQueryTotal	Esta métrica indica el total de la partición exportada.
<nombre-canalización>.document db.streamRecordsSuccessTotal	Esta métrica indica el número de registros del flujo procesados correctamente.

Alarma de CloudWatch	Descripción
<nombre-canalización> .document db.streamRecordsFailedTotal	Esta métrica indica el número total de registros del flujo que no se han podido procesar.

Uso de una canalización de OpenSearch Ingestion con Confluent Cloud Kafka

Puede utilizar una canalización de OpenSearch Ingestion para transmitir datos desde los clústeres de Confluent Cloud Kafka a los dominios de Amazon OpenSearch Service y a las colecciones de OpenSearch sin servidor. OpenSearch Ingestion admite configuraciones de redes públicas y privadas para la transmisión de datos desde los clústeres de Confluent Cloud Kafka a dominios o colecciones gestionados por OpenSearch Service u OpenSearch sin servidor.

Conectividad con clústeres de Confluent Cloud Kafka públicos

Puede usar las canalizaciones de OpenSearch Ingestion para migrar datos desde un clúster de Confluent Cloud Kafka con una configuración pública, lo que significa que el nombre de DNS del dominio se puede resolver públicamente. Para ello, configure una canalización de OpenSearch Ingestion con un clúster de Confluent Cloud Kafka público como origen y OpenSearch Service u OpenSearch sin servidor como destino. Esto procesa los datos de streaming desde un clúster de origen autogestionado a un dominio o colección de destino administrado de AWS.

Requisitos previos

Antes de crear su canalización de OpenSearch Ingestion, realice los siguientes pasos:

1. Cree un clúster de clústeres de Confluent Cloud Kafka que actúe como origen. El clúster debe contener los datos que desea ingerir en OpenSearch Service.
2. Cree un dominio de OpenSearch Service o una colección de OpenSearch sin servidor a la que desee migrar datos. Para obtener más información, consulte the section called “ Creación de dominios de OpenSearch Service” y the section called “Creación de colecciones”.
3. Configure la autenticación en el clúster de Confluent Cloud Kafka con AWS Secrets Manager. Siga los pasos de Rotate AWS Secrets Manager secrets para activar la rotación de secretos.
4. Adjunte una política basada en recursos al dominio o una política de acceso a datos a la colección. Estas políticas de acceso permiten a OpenSearch Ingestion escribir datos de su clúster autogestionado en su dominio o colección.

El siguiente ejemplo de política de acceso al dominio permite que el rol de canalización, que creará en el paso siguiente, escriba datos en un dominio. Asegúrese de actualizar el `resource` con su propio ARN.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::444455556666:role/pipeline-role"
      },
      "Action": [
        "es:DescribeDomain",
        "es:ESHttp*"
      ],
      "Resource": [
        "arn:aws:es:us-east-1:111122223333:domain/domain-name"
      ]
    }
  ]
}
```

Para crear un rol de IAM con los permisos correctos para acceder a los datos de escritura de la colección o dominio, consulte the section called “Configuración de roles y usuarios”.

Paso 1: configurar el rol de canalización

Una vez establecidos los requisitos previos de la canalización de clústeres de Confluent Cloud Kafka, configure el rol de canalización que quiere usar en la configuración de la canalización y agregue permiso para escribir en un dominio de OpenSearch Service o en una colección de OpenSearch sin servidor, así como permiso para leer los secretos de Secrets Manager.

Se necesita el siguiente permiso para administrar la interfaz de red:

JSON

```
{
```

```
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:AttachNetworkInterface",
                "ec2:CreateNetworkInterface",
                "ec2:CreateNetworkInterfacePermission",
                "ec2:DeleteNetworkInterface",
                "ec2:DeleteNetworkInterfacePermission",
                "ec2:DetachNetworkInterface",
                "ec2:DescribeNetworkInterfaces"
            ],
            "Resource": [
                "arn:aws:ec2:us-east-1:111122223333:network-interface/*",
                "arn:aws:ec2:us-east-1:111122223333:subnet/*",
                "arn:aws:ec2:us-east-1:111122223333:security-group/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeDhcpOptions",
                "ec2:DescribeRouteTables",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeSubnets",
                "ec2:DescribeVpcs",
                "ec2:Describe*"
            ],
            "Resource": "arn:aws:ec2:us-east-1:111122223333:subnet/*"
        },
        {
            "Effect": "Allow",
            "Action": [ "ec2:CreateTags" ],
            "Resource": "arn:aws:ec2:us-east-1:111122223333:network-interface/*",
            "Condition": {
               "StringEquals": { "aws:RequestTag/OSISManaged": "true" }
            }
        }
    ]
}
```

Se necesita el siguiente permiso para leer los secretos del servicio AWS Secrets Manager:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "SecretsManagerReadAccess",
            "Effect": "Allow",
            "Action": ["secretsmanager:GetSecretValue"],
            "Resource": ["arn:aws:secretsmanager:us-
east-1:111122223333:secret:,secret-name"]
        }
    ]
}
```

Se necesitan los siguientes permisos para escribir en un dominio de Amazon OpenSearch Service:

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::account-id:role/pipeline-role"
      },
      "Action": ["es:DescribeDomain", "es:ESHttp*"],
      "Resource": "arn:aws:es:region:account-id:domain/domain-name/*"
    }
  ]
}
```

Paso 2: crear la canalización

A continuación, puede configurar una canalización de OpenSearch Ingestion como la siguiente, que especifica Confluent Cloud Kafka como origen.

Puede especificar varios dominios de OpenSearch Service como destinos para sus datos. Esta capacidad permite el enrutamiento condicional o la replicación de datos entrantes en varios dominios de OpenSearch Service.

También puede migrar datos de un clúster de Confluent Kafka a una colección de VPC de OpenSearch sin servidor. Asegúrese de proporcionar una política de acceso a la red en la

configuración de la canalización. Puede utilizar un registro de esquemas de Confluent para definir un esquema de Confluent.

```
version: "2"
kafka-pipeline:
  source:
    kafka:
      encryption:
        type: "ssl"
      topics:
        - name: "topic-name"
          group_id: "group-id"
      bootstrap_servers:
        - "bootstrap-server.us-east-1.aws.private.confluent.cloud:9092"
      authentication:
        sasl:
          plain:
            username: ${aws_secrets:confluent-kafka-secret:username}
            password: ${aws_secrets:confluent-kafka-secret:password}
      schema:
        type: confluent
        registry_url: https://my-registry.us-east-1.aws.confluent.cloud
        api_key: "${{aws_secrets:schema-secret:schema_registry_api_key}}"
        api_secret: "${{aws_secrets:schema-secret:schema_registry_api_secret}}"
        basic_auth_credentials_source: "USER_INFO"
  sink:
  - opensearch:
      hosts: ["https://search-mydomain.us-east-1.es.amazonaws.com"]
      aws:
          region: "us-east-1"
  aws:
    secrets:
      confluent-kafka-secret:
        secret_id: "my-kafka-secret"
        region: "us-east-1"
      schema-secret:
        secret_id: "my-self-managed-kafka-schema"
        region: "us-east-1"
```

Puede utilizar un esquema preconfigurado para crear esta canalización. Para obtener más información, consulte the section called “Uso de esquemas”.

Conectividad con clústeres de Confluent Cloud Kafka en una VPC

También puede utilizar las canalizaciones de OpenSearch Ingestion para migrar datos desde un clúster de Confluent Cloud Kafka que se ejecuta en una VPC. Para ello, configure una canalización de OpenSearch Ingestion con un clúster de Confluent Cloud Kafka como origen y OpenSearch Service u OpenSearch sin servidor como destino. Esto procesa los datos de streaming desde un clúster fuente de Confluent Cloud Kafka a un dominio o colección de destino administrado de AWS.

OpenSearch Ingestion es compatible con los clústeres de Confluent Cloud Kafka configurados en todos los modos de red compatibles en Confluent. OpenSearch Ingestion admite los siguientes modos de configuración de red como origen:

- Emparejamiento de VPC de AWS
- AWS PrivateLink para clústeres dedicados
- AWS PrivateLink para clústeres empresariales
- AWS Transit Gateway

Requisitos previos

Antes de crear su canalización de OpenSearch Ingestion, realice los siguientes pasos:

1. Cree un clúster de Confluent Cloud Kafka con una configuración de red de VPC que contenga los datos que desee incorporar a OpenSearch Service.
2. Cree un dominio de OpenSearch Service o una colección de OpenSearch sin servidor a la que desee migrar datos. Para obtener más información, consulte the section called “ Creación de dominios de OpenSearch Service” ythe section called “Creación de colecciones”.
3. Configure la autenticación en el clúster de Confluent Cloud Kafka con AWS Secrets Manager. Siga los pasos de Rotate AWS Secrets Manager secrets para activar la rotación de secretos.
4. Obtenga el ID de la VPC que tiene acceso al clúster de Kafka de Confluent Cloud. Elija el CIDR de la VPC que utilizará OpenSearch Ingestion.

> **Note**
>
> Si utiliza la Consola de administración de AWS para crear la canalización, también debe adjuntar la canalización de OpenSearch Ingestion a la VPC para poder usar el clúster de Confluent Cloud Kafka. Para ello, busque la sección Configuración de la red, seleccione la casilla Adjuntar a la VPC y elija el CIDR de una de las opciones predeterminadas

proporcionadas o seleccione una opción propia. Puede usar cualquier CIDR de un espacio de direcciones privado, tal como se define en RFC 1918 Best Current Practice.
Para proporcionar un CIDR personalizado, seleccione Otros en el menú desplegable.
Para evitar una colisión de direcciones IP entre OpenSearch Ingestion y OpenSearch autogestionado, asegúrese de que el CIDR de la VPC de OpenSearch autogestionado sea diferente del CIDR de OpenSearch Ingestion.

5. Adjunte una política basada en recursos al dominio o una política de acceso a datos a la colección. Estas políticas de acceso permiten a OpenSearch Ingestion escribir datos de su clúster autogestionado en su dominio o colección.

> **Note**
>
> Si utiliza AWS PrivateLink para conectar Confluent Cloud Kafka, tendrá que configurar Opciones de DHCP de VPC. Nombres de host de DNS y Resolución de DNS deben estar habilitados.
> En concreto, utilice los siguientes valores de conjunto de opciones:
> Clústeres empresariales:
>
> ```
> domain-name: aws.private.confluent.cloud
> domain-name-servers: AmazonProvidedDNS
> ```
>
> Clústeres dedicados:
>
> ```
> domain-name: aws.confluent.cloud
> domain-name-servers: AmazonProvidedDNS
> ```
>
> Este cambio garantiza que la resolución de DNS para el punto de conexión de PrivateLink de Confluent funcione correctamente en la VPC.

El siguiente ejemplo de política de acceso al dominio permite que el rol de canalización, que creará en el paso siguiente, escriba datos en un dominio. Asegúrese de actualizar el `resource` con su propio ARN.

JSON

```
{
```

```
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::444455556666:role/pipeline-role"
      },
      "Action": [
        "es:DescribeDomain",
        "es:ESHttp*"
      ],
      "Resource": [
        "arn:aws:es:us-east-1:111122223333:domain/domain-name"
      ]
    }
  ]
}
```

Para crear un rol de IAM con los permisos correctos para acceder a los datos de escritura de la colección o dominio, consulte the section called “Configuración de roles y usuarios”.

Paso 1: configurar el rol de canalización

Una vez configurados los requisitos previos de la canalización, configure el rol de la canalización que desee usar en la configuración de la canalización y agregue los siguientes permisos al rol:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "SecretsManagerReadAccess",
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue"
            ],
            "Resource": ["arn:aws:secretsmanager:us-
east-1:111122223333:secret:secret-name"]
        },
        {
```

```
            "Effect": "Allow",
            "Action": [
                "ec2:AttachNetworkInterface",
                "ec2:CreateNetworkInterface",
                "ec2:CreateNetworkInterfacePermission",
                "ec2:DeleteNetworkInterface",
                "ec2:DeleteNetworkInterfacePermission",
                "ec2:DetachNetworkInterface",
                "ec2:DescribeNetworkInterfaces"
            ],
            "Resource": [
                "arn:aws:ec2:*:*:network-interface/*",
                "arn:aws:ec2:*:*:subnet/*",
                "arn:aws:ec2:*:*:security-group/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeDhcpOptions",
                "ec2:DescribeRouteTables",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeSubnets",
                "ec2:DescribeVpcs",
                "ec2:Describe*"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateTags"
            ],
            "Resource": "arn:aws:ec2:*:*:network-interface/*",
            "Condition": {
               "StringEquals":
                    {
                        "aws:RequestTag/OSISManaged": "true"
                    }
            }
        }
    ]
}
```

Debe proporcionar los permisos de Amazon EC2 anteriores en el rol de IAM que utilice para crear la canalización de OpenSearch Ingestion, ya que la canalización utiliza estos permisos para crear y eliminar una interfaz de red en la VPC. La canalización solo puede acceder al clúster de Kafka a través de esta interfaz de red.

Paso 2: crear la canalización

A continuación, puede configurar una canalización de OpenSearch Ingestion como la siguiente, que especifica Kafka como origen.

Puede especificar varios dominios de OpenSearch Service como destinos para sus datos. Esta capacidad permite el enrutamiento condicional o la replicación de datos entrantes en varios dominios de OpenSearch Service.

También puede migrar datos de un clúster de Confluent Kafka a una colección de VPC de OpenSearch sin servidor. Asegúrese de proporcionar una política de acceso a la red en la configuración de la canalización. Puede utilizar un registro de esquemas de Confluent para definir un esquema de Confluent.

```
 version: "2"
kafka-pipeline:
  source:
    kafka:
      encryption:
        type: "ssl"
      topics:
        - name: "topic-name"
          group_id: "group-id"
      bootstrap_servers:
        - "bootstrap-server.us-east-1.aws.private.confluent.cloud:9092"
      authentication:
        sasl:
          plain:
            username: ${aws_secrets:confluent-kafka-secret:username}
            password: ${aws_secrets:confluent-kafka-secret:password}
      schema:
        type: confluent
        registry_url: https://my-registry.us-east-1.aws.confluent.cloud
        api_key: "${{aws_secrets:schema-secret:schema_registry_api_key}}"
        api_secret: "${{aws_secrets:schema-secret:schema_registry_api_secret}}"
        basic_auth_credentials_source: "USER_INFO"
  sink:
```

```
  - opensearch:
      hosts: ["https://search-mydomain.us-east-1.es.amazonaws.com"]
      aws:
          region: "us-east-1"
      index: "confluent-index"
extension:
  aws:
    secrets:
      confluent-kafka-secret:
        secret_id: "my-kafka-secret"
        region: "us-east-1"
      schema-secret:
        secret_id: "my-self-managed-kafka-schema"
        region: "us-east-2"
```

Uso de una canalización de OpenSearch Ingestion con Amazon Managed Streaming for Apache Kafka

Puede utilizar el complemento de Kafka para incorporar datos de Amazon Managed Streaming para Apache Kafka (Amazon MSK) en su canalización de OpenSearch Ingestion. Con Amazon MSK, puede crear y ejecutar aplicaciones que utilizan Apache Kafka para procesar datos de streaming. OpenSearch Ingestion utiliza AWS PrivateLink para conectarse con Amazon MSK. Puede ingerir datos de clústeres de Amazon MSK y Amazon MSK sin servidor. La única diferencia entre los dos procesos son los pasos de requisitos previos que debe seguir antes de configurar su canalización.

Temas

- Requisitos previos de Amazon MSK aprovisionado
- Requisitos previos de Amazon MSK sin servidor
- Paso 1: Configurar un rol de canalización
- Paso 2: crear la canalización
- Paso 3: (opcional) utilizar AWS Glue Schema Registry
- Paso 4: (opcional) configurar las unidades de cómputo (OCU) recomendadas para la canalización de Amazon MSK

Requisitos previos de Amazon MSK aprovisionado

Antes de crear su canalización de OpenSearch Ingestion, realice los siguientes pasos:

1. Siga los pasos que se indican en Creating a cluster en la Guía para desarrolladores de Amazon Managed Streaming para Apache Kafka para crear un clúster aprovisionado de Amazon MSK. En Tipo de agente, elija cualquier opción excepto para los tipos `t3`, ya que no son compatibles con OpenSearch Ingestion.
2. Cuando el clúster tenga el estado Activo, siga los pasos que se indican en Activar la conectividad de varias VPC.
3. Siga los pasos que se indican en Adjuntar una política de clúster al clúster de MSK para adjuntar una de las siguientes políticas, en función de si el clúster y la canalización están en la misma Cuenta de AWS. Esta política permite a OpenSearch Ingestion crear una conexión AWS PrivateLink con su clúster de Amazon MSK y leer datos de los temas de Kafka. Asegúrese de actualizar el `resource` con su propio ARN.

   Las siguientes políticas se aplican cuando el clúster y la canalización están en la misma Cuenta de AWS:

   JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "osis.amazonaws.com"
      },
      "Action": [
        "kafka:CreateVpcConnection",
        "kafka:DescribeClusterV2"
      ],
      "Resource": "arn:aws:kafka:us-east-1:111122223333:cluster/cluster-
name/cluster-id"
    },
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "osis-pipelines.amazonaws.com"
      },
      "Action": [
        "kafka:CreateVpcConnection",
        "kafka:GetBootstrapBrokers",
        "kafka:DescribeClusterV2"
```

```
      ],
      "Resource": "arn:aws:kafka:us-east-1:111122223333:cluster/cluster-
name/cluster-id"
    }
  ]
}
```

Si su clúster de Amazon MSK está en una Cuenta de AWS diferente a la de su canalización, adjunte la siguiente política en su lugar. Tenga en cuenta que el acceso entre cuentas solo es posible con los clústeres de Amazon MSK aprovisionados y no con los clústeres de Amazon MSK sin servidor. El ARN para la AWS de `principal` debe ser el ARN del mismo rol de canalización que proporcionó a la configuración de YAML de la canalización:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "osis.amazonaws.com"
      },
      "Action": [
        "kafka:CreateVpcConnection",
        "kafka:DescribeClusterV2"
      ],
      "Resource": "arn:aws:kafka:us-east-1:111122223333:cluster/cluster-
name/cluster-id"
    },
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "osis-pipelines.amazonaws.com"
      },
      "Action": [
        "kafka:CreateVpcConnection",
        "kafka:GetBootstrapBrokers",
        "kafka:DescribeClusterV2"
      ],
      "Resource": "arn:aws:kafka:us-east-1:111122223333:cluster/cluster-
name/cluster-id"
    },
```

```
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::444455556666:role/pipeline-role"
      },
      "Action": [
        "kafka-cluster:*",
        "kafka:*"
      ],
      "Resource": [
        "arn:aws:kafka:us-east-1:111122223333:cluster/cluster-name/cluster-id",
        "arn:aws:kafka:us-east-1:111122223333:topic/cluster-name/cluster-id/*",
        "arn:aws:kafka:us-east-1:111122223333:group/cluster-name/*"
      ]
    }
  ]
}
```

4. Cree un tema de Kafka siguiendo los pasos de Creación de un tema. Asegúrese de que *BootstrapServerString* sea una de las URL de arranque del punto de conexión privado (VPC única). El valor de `--replication-factor` debe ser 2 o 3, en función del número de zonas que tenga su clúster de Amazon MSK. El valor de las `--partitions` debe ser como mínimo `1`.
5. Produzca y consuma datos siguiendo los pasos de Producir y consumir datos. De nuevo, asegúrese de que *BootstrapServerString* sea una de las URL de arranque del punto de conexión privado (VPC única).

Requisitos previos de Amazon MSK sin servidor

Antes de crear su canalización de OpenSearch Ingestion, realice los siguientes pasos:

1. Siga los pasos que se indican en Create an MSK Serverless cluster en la Guía para desarrolladores de Amazon Managed Streaming para Apache Kafka para crear un clúster de Amazon MSK sin servidor.
2. Cuando el clúster tenga el estado Activo, siga los pasos que se indican en Attach a cluster policy to the MSK cluster para adjuntar la siguiente política. Asegúrese de actualizar el `resource` con su propio ARN.

   JSON

```
{
```

```
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "osis.amazonaws.com"
      },
      "Action": [
        "kafka:CreateVpcConnection",
        "kafka:DescribeClusterV2"
      ],
      "Resource": "arn:aws:kafka:us-east-1:111122223333:cluster/cluster-
name/cluster-id"
    },
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "osis-pipelines.amazonaws.com"
      },
      "Action": [
        "kafka:CreateVpcConnection",
        "kafka:GetBootstrapBrokers",
        "kafka:DescribeClusterV2"
      ],
      "Resource": "arn:aws:kafka:us-east-1:111122223333:cluster/cluster-
name/cluster-id"
    }
  ]
}
```

Esta política permite a OpenSearch Ingestion crear una conexión de AWS PrivateLink con su clúster de Amazon MSK sin servidor y leer datos de los temas de Kafka. Esta política se aplica cuando el clúster y la canalización se encuentran en la misma Cuenta de AWS, lo que debe ser cierto, ya que Amazon MSK sin servidor no admite el acceso entre cuentas.

3. Cree un tema de Kafka siguiendo los pasos de Creación de un tema. Asegúrese de que *BootstrapServerString* sea una de las URL de arranque de IAM de su capa de seguridad y autenticación simple (SASL). El valor de `--replication-factor` debe ser 2 o 3, en función del número de zonas que tenga su clúster de Amazon MSK sin servidor. El valor de las `--partitions` debe ser como mínimo `10`.

4. Produzca y consuma datos siguiendo los pasos de Producir y consumir datos. De nuevo, asegúrese de que *BootstrapServerString* sea una de las URL de arranque de IAM de su capa de seguridad y autenticación simple (SASL).

Paso 1: Configurar un rol de canalización

Una vez configurado el clúster aprovisionado o sin servidor de Amazon MSK, agregue los siguientes permisos de Kafka al rol de canalización que desee usar en la configuración de la canalización:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "kafka-cluster:Connect",
                "kafka-cluster:AlterCluster",
                "kafka-cluster:DescribeCluster",
                "kafka:DescribeClusterV2",
                "kafka:GetBootstrapBrokers"
            ],
            "Resource": [
                "arn:aws:kafka:us-east-1:111122223333:cluster/cluster-
name/cluster-id"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "kafka-cluster:*Topic*",
                "kafka-cluster:ReadData"
            ],
            "Resource": [
                "arn:aws:kafka:us-east-1:111122223333:topic/cluster-name/cluster-
id/topic-name"
            ]
        },
        {
            "Effect": "Allow",
```

```
            "Action": [
                "kafka-cluster:AlterGroup",
                "kafka-cluster:DescribeGroup"
            ],
            "Resource": [
                "arn:aws:kafka:us-east-1:111122223333:group/cluster-name/*"
            ]
        }
    ]
}
```

Paso 2: crear la canalización

A continuación, puede configurar una canalización de OpenSearch Ingestion como la siguiente, en la que se especifica a Kafka como fuente:

```
version: "2"
log-pipeline:
  source:
    kafka:
      acknowledgements: true
      topics:
      - name: "topic-name"
        group_id: "grouplambd-id"
      aws:
        msk:
          arn: "arn:aws:kafka:region:account-id:cluster/cluster-name/cluster-id"
        region: "us-west-2"
  processor:
  - grok:
      match:
        message:
        - "%{COMMONAPACHELOG}"
  - date:
      destination: "@timestamp"
      from_time_received: true
  sink:
  - opensearch:
      hosts: ["https://search-domain-endpoint.us-east-1es.amazonaws.com"]
      index: "index_name"
      aws_region: "region"
      aws_sigv4: true
```

Puede utilizar un esquema de Amazon MSK preconfigurado para crear esta canalización. Para obtener más información, consulte the section called “Uso de esquemas”.

Paso 3: (opcional) utilizar AWS Glue Schema Registry

Al utilizar OpenSearch Ingestion con Amazon MSK, puede utilizar el formato de datos AVRO para los esquemas alojados en AWS Glue Schema Registry. Con AWS Glue Schema Registry, puede descubrir, controlar y evolucionar de forma centralizada los esquemas de flujo de datos.

Para usar esta opción, habilite el `type` de esquema en la configuración de su canalización:

```
schema:
  type: "aws_glue"
```

También debe proporcionar a AWS Glue permisos de acceso de lectura en su rol de canalización. Puede usar la política administrada por AWS denominada AWSGlueSchemaRegistryReadonlyAccess. Además, su registro debe estar en la misma Cuenta de AWS y región que su canalización de OpenSearch Ingestion.

Paso 4: (opcional) configurar las unidades de cómputo (OCU) recomendadas para la canalización de Amazon MSK

Cada unidad de cómputo tiene un consumidor por tema. Los agentes equilibran las particiones entre estos consumidores para un tema determinado. Sin embargo, cuando el número de particiones es superior al número de consumidores, Amazon MSK aloja varias particiones en cada consumidor. OpenSearch Ingestion tiene un escalado automático integrado para escalar o reducir verticalmente según el uso de la CPU o la cantidad de registros pendientes en la canalización.

Para un rendimiento óptimo, distribuya las particiones entre varias unidades de cómputo para el procesamiento paralelo. Si los temas tienen un gran número de particiones (por ejemplo, más de 96, que es el número máximo de OCU por canalización), se recomienda configurar una canalización con 1–96 OCU. Esto se debe a que se escalará automáticamente según sea necesario. Si un tema tiene un número reducido de particiones (por ejemplo, menos de 96), mantenga la unidad de cómputo máxima igual a la cantidad de particiones.

Cuando una canalización tenga más de un tema, seleccione el tema con el mayor número de particiones como referencia para configurar las unidades de cómputo máximas. Al agregar otra canalización con un nuevo conjunto de OCU al mismo tema y grupo de consumidores, se puede escalar el rendimiento de forma casi lineal.

Uso de una canalización de OpenSearch Ingestion con Amazon RDS

Puede usar una canalización de OpenSearch Ingestion con Amazon RDS para exportar los datos existentes y transmitir los cambios (como crear, actualizar y eliminar) a los dominios y recopilaciones de Amazon OpenSearch Service. La canalización de OpenSearch Ingestion incorpora una infraestructura de captura de datos de cambios (CDC) para proporcionar una forma a gran escala y de baja latencia de transmitir datos de forma continua desde Amazon RDS. Se admiten RDS para MySQL y RDS para PostgreSQL.

Existen dos formas de utilizar Amazon RDS como origen para procesar datos: con o sin una instantánea inicial completa. Una instantánea inicial completa es una instantánea de las tablas especificadas y esta instantánea se exporta a Amazon S3. Desde allí, una canalización de OpenSearch Ingestion la envía a un índice de un dominio o la particiona en varios índices en un dominio. Para mantener la coherencia de los datos en Amazon RDS y OpenSearch, la canalización sincroniza todos los eventos de creación, actualización y eliminación de la tabla de Amazon RDS con los documentos guardados en el índice o los índices de OpenSearch.

Cuando utiliza una instantánea inicial completa, la canalización de OpenSearch Ingestion ingiere primero la instantánea y, a continuación, comienza a leer los datos de los flujos de cambios de Amazon RDS. Con el tiempo, recupera y mantiene la coherencia de datos casi en tiempo real entre Amazon RDS y OpenSearch.

También puede usar la integración de OpenSearch Ingestion con Amazon RDS para realizar un seguimiento de los cambios, capturar datos e ingerir todas las actualizaciones de Aurora en OpenSearch. Elija esta opción si ya tiene una instantánea completa de algún otro mecanismo o si solo quiere capturar todos los cambios en los datos de una instancia de Amazon RDS.

Si elige esta opción, debe configurar el registro binario de Amazon RDS para MySQL o configurar la replicación lógica para la instancia de base de datos Amazon RDS para PostgresSQL.

Temas

- RDS for MySQL
- RDS para PostgreSQL

RDS for MySQL

Realice los siguientes pasos para configurar una canalización de OpenSearch Ingestion con Amazon RDS para RDS for MySQL.

Temas

- Requisitos previos de RDS para MySQL
- Paso 1: configurar el rol de canalización
- Paso 2: crear la canalización
- Coherencia de datos
- Asignación de tipos de datos
- Limitaciones
- Alarmas de CloudWatch recomendadas

Requisitos previos de RDS para MySQL

Antes de crear su canalización de OpenSearch Ingestion, realice los siguientes pasos:

1. Cree un grupo de parámetros de base de datos personalizado en Amazon RDS para configurar el registro binario y establecer los siguientes parámetros.

   ```
   binlog_format=ROW
   binlog_row_image=full
   binlog_row_metadata=FULL
   ```

   Compruebe también que el parámetro `binlog_row_value_options` no esté establecido en `PARTIAL_JSON`.

   Para obtener más información, consulte Configuración del registro binario de RDS para MySQL.
2. Seleccione o cree una instancia de base de datos RDS para MySQL y asocie el grupo de parámetros creado en el paso anterior a la instancia de base de datos.
3. Compruebe que las copias de seguridad automáticas están activadas en la base de datos. Para obtener más información, consulte Habilitación de copias de seguridad automatizadas.
4. Configure la retención de registros binarios con tiempo suficiente para que se produzca la replicación, por ejemplo, 24 horas. Para obtener más información, consulte Configuración y visualización de la configuración de registros binarios en la Guía del usuario de Amazon RDS.
5. Configure la autenticación de nombre de usuario y contraseña en su instancia de Amazon RDS mediante la administración de contraseñas con Amazon RDS y AWS Secrets Manager. También puede crear una combinación de nombre de usuario y contraseña mediante la creación de un secreto de Secrets Manager.

6. Si utiliza la característica de instantáneas iniciales completa, cree una AWS KMS key y un rol de IAM para exportar datos de Amazon RDS a Amazon S3.

   El rol de IAM debe tener la siguiente política de permisos:

   JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "ExportPolicy",
            "Effect": "Allow",
            "Action": [
                "s3:PutObject*",
                "s3:ListBucket",
                "s3:GetObject*",
                "s3:DeleteObject*",
                "s3:GetBucketLocation"
            ],
            "Resource": [
                "arn:aws:s3:::s3-bucket-used-in-pipeline",
                "arn:aws:s3:::s3-bucket-used-in-pipeline/*"
            ]
        }
    ]
}
```

   El rol también debe tener las siguientes relaciones de confianza:

   JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "export.rds.amazonaws.com"
            },
            "Action": "sts:AssumeRole"
        }
    ]
```

```
}
```

7. Seleccione o cree un dominio de OpenSearch Service o una colección de OpenSearch sin servidor. Para obtener más información, consulte Creating OpenSearch Service domains y Creating collections.
8. Adjunte una política basada en recursos al dominio o una política de acceso a datos a la colección. Estas políticas de acceso permiten a OpenSearch Ingestion escribir datos de su instancia de base de datos de Amazon RDS en su dominio o colección.

Paso 1: configurar el rol de canalización

Una vez configurados los requisitos previos de la canalización de Amazon RDS, configure el rol de la canalización que se usará en la configuración de la canalización. Agregue también los siguientes permisos para el origen de Amazon RDS al rol:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
    {
    "Sid": "allowReadingFromS3Buckets",
    "Effect": "Allow",
    "Action": [
    "s3:GetObject",
    "s3:DeleteObject",
    "s3:GetBucketLocation",
    "s3:ListBucket",
    "s3:PutObject"
    ],
    "Resource": [
    "arn:aws:s3:::s3_bucket",
    "arn:aws:s3:::s3_bucket/*"
    ]
    },
    {
    "Sid": "allowNetworkInterfacesActions",
    "Effect": "Allow",
    "Action": [
    "ec2:AttachNetworkInterface",
    "ec2:CreateNetworkInterface",
```

```
    "ec2:CreateNetworkInterfacePermission",
    "ec2:DeleteNetworkInterface",
    "ec2:DeleteNetworkInterfacePermission",
    "ec2:DetachNetworkInterface",
    "ec2:DescribeNetworkInterfaces"
    ],
    "Resource": [
    "arn:aws:ec2:*:111122223333:network-interface/*",
    "arn:aws:ec2:*:111122223333:subnet/*",
    "arn:aws:ec2:*:111122223333:security-group/*"
    ]
    },
    {
    "Sid": "allowDescribeEC2",
    "Effect": "Allow",
    "Action": [
    "ec2:Describe*"
    ],
    "Resource": "*"
    },
    {
    "Sid": "allowTagCreation",
    "Effect": "Allow",
    "Action": [
    "ec2:CreateTags"
    ],
    "Resource": "arn:aws:ec2:*:111122223333:network-interface/*",
    "Condition": {
    "StringEquals": {
    "aws:RequestTag/OSISManaged": "true"
    }
    }
    },
    {
    "Sid": "AllowDescribeInstances",
    "Effect": "Allow",
    "Action": [
    "rds:DescribeDBInstances"
    ],
    "Resource": [
    "arn:aws:rds:us-east-2:111122223333:db:*"
    ]
    },
    {
```

```
    "Sid": "AllowSnapshots",
    "Effect": "Allow",
    "Action": [
    "rds:DescribeDBSnapshots",
    "rds:CreateDBSnapshot",
    "rds:AddTagsToResource"
    ],
    "Resource": [
    "arn:aws:rds:us-east-2:111122223333:db:DB-id",
    "arn:aws:rds:us-east-2:111122223333:snapshot:DB-id*"
    ]
    },
    {
    "Sid": "AllowExport",
    "Effect": "Allow",
    "Action": [
    "rds:StartExportTask"
    ],
    "Resource": [
    "arn:aws:rds:us-east-2:111122223333:snapshot:DB-id*"
    ]
    },
    {
    "Sid": "AllowDescribeExports",
    "Effect": "Allow",
    "Action": [
    "rds:DescribeExportTasks"
    ],
    "Resource": "*",
    "Condition": {
    "StringEquals": {
    "aws:RequestedRegion": "us-east-2",
    "aws:ResourceAccount": "111122223333"
    }
    }
    },
    {
    "Sid": "AllowAccessToKmsForExport",
    "Effect": "Allow",
    "Action": [
    "kms:Decrypt",
    "kms:Encrypt",
    "kms:DescribeKey",
    "kms:RetireGrant",
```

```
    "kms:CreateGrant",
    "kms:ReEncrypt*",
    "kms:GenerateDataKey*"
    ],
    "Resource": [
    "arn:aws:kms:us-east-2:111122223333:key/export-key-id"
    ]
    },
    {
    "Sid": "AllowPassingExportRole",
    "Effect": "Allow",
    "Action": "iam:PassRole",
    "Resource": [
    "arn:aws:iam::111122223333:role/export-role"
    ]
    },
    {
    "Sid": "SecretsManagerReadAccess",
    "Effect": "Allow",
    "Action": [
    "secretsmanager:GetSecretValue"
    ],
    "Resource": [
    "arn:aws:secretsmanager:*:111122223333:secret:*"
    ]
    }
    ]
    }
```

Paso 2: crear la canalización

Configure una canalización de OpenSearch Ingestion similar a la siguiente. La canalización de ejemplo especifica una instancia de Amazon RDS como origen.

```
version: "2"
rds-mysql-pipeline:
  source:
    rds:
      db_identifier: "instance-id"
      engine: mysql
      database: "database-name"
      tables:
```

```
        include:
          - "table1"
          - "table2"
      s3_bucket: "bucket-name"
      s3_region: "bucket-region"
      s3_prefix: "prefix-name"
      export:
        kms_key_id: "kms-key-id"
        iam_role_arn: "export-role-arn"
      stream: true
      aws:
        sts_role_arn: "arn:aws:iam::account-id:role/pipeline-role"
        region: "us-east-1"
      authentication:
        username: ${{aws_secrets:secret:username}}
        password: ${{aws_secrets:secret:password}}
  sink:
    - opensearch:
        hosts: ["https://search-mydomain.us-east-1.es.amazonaws.com"]
        index: "${getMetadata(\"table_name\")}"
        index_type: custom
        document_id: "${getMetadata(\"primary_key\")}"
        action: "${getMetadata(\"opensearch_action\")}"
        document_version: "${getMetadata(\"document_version\")}"
        document_version_type: "external"
        aws:
          sts_role_arn: "arn:aws:iam::account-id:role/pipeline-role"
          region: "us-east-1"
extension:
  aws:
    secrets:
      secret:
        secret_id: "rds-secret-id"
        region: "us-east-1"
        sts_role_arn: "arn:aws:iam::account-id:role/pipeline-role"
        refresh_interval: PT1H
```

Puede utilizar un esquema de Amazon RDS preconfigurado para crear esta canalización. Para obtener más información, consulte Uso de esquemas.

A fin de utilizar Amazon Aurora como origen, debe configurar el acceso a la VPC para la canalización. La VPC que elija debe ser la misma que usa su origen de Amazon Aurora. A continuación, elija una o varias subredes y uno o varios grupos de seguridad de VPC. Tenga en

cuenta que la canalización necesita acceso de red a una base de datos Aurora MySQL, por lo que también debe comprobar que el clúster de Aurora esté configurado con un grupo de seguridad de VPC que permita el tráfico entrante desde el grupo de seguridad de VPC de la canalización al puerto de la base de datos. Para obtener más información, consulte Control de acceso con grupos de seguridad.

Si utiliza la Consola de administración de AWS para crear la canalización, también debe adjuntar la canalización a la VPC para poder usar Amazon Aurora como origen. Para ello, busque la sección Configuración de la red, seleccione Adjuntar a la VPC y elija el CIDR de una de las opciones predeterminadas proporcionadas o seleccione una opción propia. Puede usar cualquier CIDR de un espacio de direcciones privado, tal como se define en RFC 1918 Best Current Practice.

Para proporcionar un CIDR personalizado, seleccione Otros en el menú desplegable. Para evitar una colisión de direcciones IP entre OpenSearch Ingestion y Amazon RDS, asegúrese de que el CIDR de la VPC de Amazon RDS sea diferente del CIDR de OpenSearch Ingestion.

Para obtener más información, consulte Configuring VPC access for a pipeline.

Coherencia de datos

La canalización garantiza la coherencia de datos mediante el sondeo continuo o la recepción de cambios de la instancia de Amazon RDS y la actualización de los documentos correspondientes en el índice de OpenSearch.

OpenSearch Ingestion admite el reconocimiento integral para garantizar la durabilidad de los datos. Cuando una canalización lee instantáneas o transmisiones, crea particiones de forma dinámica para el procesamiento paralelo. La canalización marca una partición como completa cuando recibe un reconocimiento después de incorporar todos los registros en el dominio o la colección de OpenSearch. Si desea incorporarlos a una colección de búsquedas de OpenSearch sin servidor, puede generar un ID de documento en la canalización. Si quiere ingerirlos en una colección de series temporales de OpenSearch sin servidor, tenga en cuenta que la canalización no genera un ID de documento, de modo que debe omitir `document_id: "${getMetadata(\"primary_key\")}"` en la configuración del receptor de la canalización.

Una canalización de OpenSearch Ingestion también asigna las acciones de los eventos entrantes a las correspondientes acciones de indexación masiva para facilitar la incorporación de documentos. Esto mantiene la coherencia de datos, de modo que cada cambio de datos en Amazon RDS se concilia con los cambios de documento correspondientes en OpenSearch.

Asignación de tipos de datos

La canalización de OpenSearch Ingestion asigna los tipos de datos de MySQL a representaciones que son adecuadas para el consumo de los dominios o las colecciones de OpenSearch Service. Si no se ha definido ninguna plantilla de asignación en OpenSearch, OpenSearch determina automáticamente los tipos de campo con una asignación dinámica en función del primer documento enviado. También puede definir de forma explícita los tipos de campo que mejor se adapten a sus necesidades en OpenSearch mediante una plantilla de asignación.

En la siguiente tabla, se muestran los tipos de datos de MySQL y los tipos de campos de OpenSearch correspondientes. La columna Tipo de campo de OpenSearch predeterminado muestra el tipo de campo correspondiente en OpenSearch si no se ha definido una asignación explícita. En este caso, OpenSearch determina automáticamente los tipos de campo mediante una asignación dinámica. La columna Tipo de campo de OpenSearch recomendado es el tipo de campo correspondiente que se recomienda especificar de forma explícita en una plantilla de asignación. Estos tipos de campos están más alineados con los tipos de datos de MySQL y, por lo general, pueden habilitar mejores características de búsqueda disponibles en OpenSearch.

Tipo de datos de MySQL	Tipo de campo de OpenSearch predeterminado	Tipo de campo de OpenSearch recomendado
BIGINT	long	long
BIGINT UNSIGNED	long	largo sin signo
BIT	long	byte, corto, entero o largo según el número de bits
DECIMAL	text	doble o palabra
DOUBLE	float	double
FLOAT	float	float
INT	long	entero

Tipo de datos de MySQL	Tipo de campo de OpenSearch predeterminado	Tipo de campo de OpenSearch recomendado
INT UNSIGNED	long	long
MEDIUMIN	long	entero
MEDIUMIN UNSIGNED	long	entero
NUMERIC	text	doble o palabra
SMALLINT	long	short
SMALLINT UNSIGNED	long	entero
TINYINT	long	byte
TINYINT UNSIGNED	long	short
BINARIO	text	binario
BLOB	text	binario
CHAR	text	text
ENUM	text	palabra clave
LONGBLOE	text	binario
LONGTEXT	text	text
MEDIUMBL B	text	binario
MEDIUMTE T	text	text

Tipo de datos de MySQL	Tipo de campo de OpenSearch predeterminado	Tipo de campo de OpenSearch recomendado
SET	text	palabra clave
TEXT	text	text
TINYBLOB	text	binario
TINYTEXT	text	text
VARBINAR	text	binario
VARCHAR	text	text
DATE	largo (en milisegundos de época)	date
DATETIME	largo (en milisegundos de época)	date
TIME	largo (en milisegundos de época)	date
TIMESTAM	largo (en milisegundos de época)	date
YEAR	largo (en milisegundos de época)	date
GEOMETR	texto (en formato WKT)	geo_shape
GEOMETR OLLECTIO	texto (en formato WKT)	geo_shape
LINESTRIN G	texto (en formato WKT)	geo_shape
MULTILINE STRING	texto (en formato WKT)	geo_shape
MULTIPOI T	texto (en formato WKT)	geo_shape

Tipo de datos de MySQL	Tipo de campo de OpenSearch predeterminado	Tipo de campo de OpenSearch recomendado
MULTIPOLYGON	texto (en formato WKT)	geo_shape
POINT	texto (en formato WKT)	geo_point o geo_shape
POLYGON	texto (en formato WKT)	geo_shape
JSON	text	objeto

Le recomendamos configurar la cola de mensajes fallidos (DLQ) en su canalización de OpenSearch Ingestion. Si ha configurado la cola, OpenSearch Service envía todos los documentos con errores que no se pueden incorporar debido a errores de asignación dinámica.

Si las asignaciones automáticas fallan, puede usar `template_type` y `template_content` en su configuración de canalización para definir reglas de asignación explícitas. Como alternativa, puede crear plantillas de asignación directamente en su dominio o colección de búsqueda antes de iniciar la canalización.

Limitaciones

Tenga en cuenta las siguientes limitaciones cuando configure una canalización de OpenSearch Ingestion para RDS para MySQL:

- La integración solo admite una base de datos MySQL por canalización.
- Actualmente, la integración no admite la ingesta de datos entre regiones; la instancia de Amazon RDS y el dominio de OpenSearch deben estar en la misma Región de AWS.
- Actualmente, la integración no admite la ingesta de datos entre cuentas; la instancia de Amazon RDS y la canalización de OpenSearch Ingestion deben estar en la misma Cuenta de AWS.
- Asegúrese de que la instancia de Amazon RDS tenga habilitada la autenticación mediante Secrets Manager, que es el único mecanismo de autenticación admitido.
- La configuración de canalización existente no se puede actualizar para ingerir datos de una base de datos diferente o de una tabla diferente. Para actualizar la base de datos o el nombre de la tabla de una canalización, debe crear una canalización nueva.

- Por lo general, las instrucciones en lenguaje de definición de datos (DDL) no son compatibles. La coherencia de datos no se mantendrá si:
  - Se cambian las claves principales (agregar/eliminar/cambiar el nombre).
  - Las tablas se eliminan o se truncan.
  - Se cambian los nombres de las columnas o los tipos de datos.
- Si las tablas MySQL que se van a sincronizar no tienen definidas las claves principales, no se garantiza la coherencia de datos. Deberá definir correctamente la opción `document_id` personalizada en la configuración del receptor de OpenSearch para poder sincronizar las actualizaciones o eliminaciones con OpenSearch.
- No se admiten las referencias a claves externas con acciones de eliminación en cascada y pueden provocar inconsistencias en los datos entre RDS for MySQL y OpenSearch.
- No se admiten los clústeres de bases de datos de zonas de disponibilidad múltiple de Amazon RDS.
- Versiones compatibles: MySQL 8.0 y versiones posteriores.

Alarmas de CloudWatch recomendadas

Se recomiendan las siguientes métricas de CloudWatch para supervisar el rendimiento de la canalización de ingesta. Estas métricas pueden ayudarlo a identificar la cantidad de datos procesados a partir de exportaciones y flujos, los errores al procesar eventos de transmisión y exportación y el número de documentos escritos en el destino. Puede configurar las alarmas de CloudWatch para que lleven a cabo una acción cuando una de estas métricas supere un valor especificado durante un periodo de tiempo determinado.

Métrica	Descripción
pipeline-name .rds.credentialsChanged	Esta métrica indica la frecuencia con la que se rotan los secretos de AWS.
pipeline-name .rds.executorRefreshErrors	Esta métrica indica errores al actualizar los secretos de AWS.

Métrica	Descripción
pipeline-name .rds.exportRecordsTotal	Esta métrica indica el número de registros exportados desde Amazon Aurora.
pipeline-name .rds.exportRecordsProcessed	Esta métrica indica el número de registros procesados por la canalización de OpenSearch Ingestion.
pipeline-name .rds.exportRecordProcessingErrors	Esta métrica indica el número de errores de procesamiento en una canalización de OpenSearch Ingestion al leer los datos de un clúster de Amazon Aurora.
pipeline-name .rds.exportRecordsSuccessTotal	Esta métrica indica el número total de registros de exportación procesados correctamente.
pipeline-name .rds.exportRecordsFailedTotal	Esta métrica indica el número total de registros que no se han podido procesar.
pipeline-name .rds.bytesReceived	Esta métrica indica el número total de bytes recibidos por una canalización de OpenSearch Ingestion.
pipeline-name .rds.bytesProcessed	Esta métrica indica el número total de bytes procesados por una canalización de OpenSearch Ingestion.
pipeline-name .rds.streamRecordsSuccessTotal	Esta métrica indica el número de registros del flujo procesados correctamente.

Métrica	Descripción
pipeline-name .rds.streamRecordsFailedTotal	Esta métrica indica el número total de registros del flujo que no se han podido procesar.

RDS para PostgreSQL

Complete los siguientes pasos para configurar una canalización de OpenSearch Ingestion con Amazon RDS para RDS para PostgreSQL.

Temas

- Requisitos previos de PostgreSQL
- Paso 1: configurar el rol de canalización
- Paso 2: crear la canalización
- Coherencia de datos
- Asignación de tipos de datos
- Limitaciones
- Alarmas de CloudWatch recomendadas

Requisitos previos de PostgreSQL

Antes de crear su canalización de OpenSearch Ingestion, realice los siguientes pasos:

1. Cree un grupo de parámetros de base de datos personalizado en Amazon RDS para configurar la replicación lógica.

   ```
   rds.logical_replication=1
   ```

   Para obtener más información, consulte Replicación lógica de Amazon RDS para PostgreSQL.
2. Seleccione o cree una instancia de base de datos de RDS para PostgreSQL y asocie el grupo de parámetros creado en el paso 1 a la instancia de base de datos.
3. Configure la autenticación de nombre de usuario y contraseña en su instancia de Amazon RDS mediante la administración de contraseñas con Aurora y AWS Secrets Manager. También puede

crear una combinación de nombre de usuario y contraseña mediante la creación de un secreto de Secrets Manager.

4. Si utiliza la característica de instantáneas iniciales completa, cree una AWS KMS key y un rol de IAM para exportar datos de Amazon RDS a Amazon S3.

   El rol de IAM debe tener la siguiente política de permisos:

   JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "ExportPolicy",
            "Effect": "Allow",
            "Action": [
                "s3:PutObject*",
                "s3:ListBucket",
                "s3:GetObject*",
                "s3:DeleteObject*",
                "s3:GetBucketLocation"
            ],
            "Resource": [
                "arn:aws:s3:::s3-bucket-used-in-pipeline",
                "arn:aws:s3:::s3-bucket-used-in-pipeline/*"
            ]
        }
    ]
}
```

   El rol también debe tener las siguientes relaciones de confianza:

   JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "export.rds.amazonaws.com"
            },
```

```
            "Action": "sts:AssumeRole"
        }
    ]
}
```

5. Seleccione o cree un dominio de OpenSearch Service o una colección de OpenSearch sin servidor. Para obtener más información, consulte Creating OpenSearch Service domains y Creating collections.
6. Adjunte una política basada en recursos al dominio o una política de acceso a datos a la colección. Estas políticas de acceso permiten a OpenSearch Ingestion escribir datos de su instancia de base de datos de Amazon RDS en su dominio o colección.

Paso 1: configurar el rol de canalización

Una vez configurados los requisitos previos de la canalización de Amazon RDS, configure el rol de la canalización que se usará en la configuración de la canalización. Agregue también los siguientes permisos para el origen de Amazon RDS al rol:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
    {
    "Sid": "allowReadingFromS3Buckets",
    "Effect": "Allow",
    "Action": [
    "s3:GetObject",
    "s3:DeleteObject",
    "s3:GetBucketLocation",
    "s3:ListBucket",
    "s3:PutObject"
    ],
    "Resource": [
    "arn:aws:s3:::s3_bucket",
    "arn:aws:s3:::s3_bucket/*"
    ]
    },
    {
    "Sid": "allowNetworkInterfacesActions",
    "Effect": "Allow",
```

```
    "Action": [
    "ec2:AttachNetworkInterface",
    "ec2:CreateNetworkInterface",
    "ec2:CreateNetworkInterfacePermission",
    "ec2:DeleteNetworkInterface",
    "ec2:DeleteNetworkInterfacePermission",
    "ec2:DetachNetworkInterface",
    "ec2:DescribeNetworkInterfaces"
    ],
    "Resource": [
    "arn:aws:ec2:*:111122223333:network-interface/*",
    "arn:aws:ec2:*:111122223333:subnet/*",
    "arn:aws:ec2:*:111122223333:security-group/*"
    ]
    },
    {
    "Sid": "allowDescribeEC2",
    "Effect": "Allow",
    "Action": [
    "ec2:Describe*"
    ],
    "Resource": "*"
    },
    {
    "Sid": "allowTagCreation",
    "Effect": "Allow",
    "Action": [
    "ec2:CreateTags"
    ],
    "Resource": "arn:aws:ec2:*:111122223333:network-interface/*",
    "Condition": {
    "StringEquals": {
    "aws:RequestTag/OSISManaged": "true"
    }
    }
    },
    {
    "Sid": "AllowDescribeInstances",
    "Effect": "Allow",
    "Action": [
    "rds:DescribeDBInstances"
    ],
    "Resource": [
    "arn:aws:rds:us-east-2:111122223333:db:*"
```

```
    ]
    },
    {
    "Sid": "AllowSnapshots",
    "Effect": "Allow",
    "Action": [
    "rds:DescribeDBSnapshots",
    "rds:CreateDBSnapshot",
    "rds:AddTagsToResource"
    ],
    "Resource": [
    "arn:aws:rds:us-east-2:111122223333:db:DB-id",
    "arn:aws:rds:us-east-2:111122223333:snapshot:DB-id*"
    ]
    },
    {
    "Sid": "AllowExport",
    "Effect": "Allow",
    "Action": [
    "rds:StartExportTask"
    ],
    "Resource": [
    "arn:aws:rds:us-east-2:111122223333:snapshot:DB-id*"
    ]
    },
    {
    "Sid": "AllowDescribeExports",
    "Effect": "Allow",
    "Action": [
    "rds:DescribeExportTasks"
    ],
    "Resource": "*",
    "Condition": {
    "StringEquals": {
    "aws:RequestedRegion": "us-east-2",
    "aws:ResourceAccount": "111122223333"
    }
    }
    },
    {
    "Sid": "AllowAccessToKmsForExport",
    "Effect": "Allow",
    "Action": [
    "kms:Decrypt",
```

```
    "kms:Encrypt",
    "kms:DescribeKey",
    "kms:RetireGrant",
    "kms:CreateGrant",
    "kms:ReEncrypt*",
    "kms:GenerateDataKey*"
    ],
    "Resource": [
    "arn:aws:kms:us-east-2:111122223333:key/export-key-id"
    ]
    },
    {
    "Sid": "AllowPassingExportRole",
    "Effect": "Allow",
    "Action": "iam:PassRole",
    "Resource": [
    "arn:aws:iam::111122223333:role/export-role"
    ]
    },
    {
    "Sid": "SecretsManagerReadAccess",
    "Effect": "Allow",
    "Action": [
    "secretsmanager:GetSecretValue"
    ],
    "Resource": [
    "arn:aws:secretsmanager:*:111122223333:secret:*"
    ]
    }
    ]
    }
```

Paso 2: crear la canalización

Configure una canalización de OpenSearch Ingestion como la siguiente, que especifica una instancia de RDS para PostgreSQL como origen.

```
version: "2"
rds-postgres-pipeline:
  source:
    rds:
      db_identifier: "instance-id"
```

```
      engine: postgresql
      database: "database-name"
      tables:
        include:
          - "schema1.table1"
          - "schema2.table2"
      s3_bucket: "bucket-name"
      s3_region: "bucket-region"
      s3_prefix: "prefix-name"
      export:
        kms_key_id: "kms-key-id"
        iam_role_arn: "export-role-arn"
      stream: true
      aws:
        sts_role_arn: "arn:aws:iam::account-id:role/pipeline-role"
        region: "us-east-1"
      authentication:
        username: ${{aws_secrets:secret:username}}
        password: ${{aws_secrets:secret:password}}
  sink:
    - opensearch:
        hosts: ["https://search-mydomain.us-east-1.es.amazonaws.com"]
        index: "${getMetadata(\"table_name\")}"
        index_type: custom
        document_id: "${getMetadata(\"primary_key\")}"
        action: "${getMetadata(\"opensearch_action\")}"
        document_version: "${getMetadata(\"document_version\")}"
        document_version_type: "external"
        aws:
          sts_role_arn: "arn:aws:iam::account-id:role/pipeline-role"
          region: "us-east-1"
extension:
  aws:
    secrets:
      secret:
        secret_id: "rds-secret-id"
        region: "us-east-1"
        sts_role_arn: "arn:aws:iam::account-id:role/pipeline-role"
        refresh_interval: PT1H
```

> **Note**
>
> Puede utilizar un esquema de Amazon RDS preconfigurado para crear esta canalización. Para obtener más información, consulte Uso de esquemas.

A fin de utilizar Amazon Aurora como origen, debe configurar el acceso a la VPC para la canalización. La VPC que elija debe ser la misma que usa su origen de Amazon Aurora. A continuación, elija una o varias subredes y uno o varios grupos de seguridad de VPC. Tenga en cuenta que la canalización necesita acceso de red a una base de datos Aurora MySQL, por lo que también debe comprobar que el clúster de Aurora esté configurado con un grupo de seguridad de VPC que permita el tráfico entrante desde el grupo de seguridad de VPC de la canalización al puerto de la base de datos. Para obtener más información, consulte Control de acceso con grupos de seguridad.

Si utiliza la Consola de administración de AWS para crear la canalización, también debe adjuntar la canalización a la VPC para poder usar Amazon Aurora como origen. Para ello, busque la sección Configuración de la red, seleccione Adjuntar a la VPC y elija el CIDR de una de las opciones predeterminadas proporcionadas o seleccione una opción propia. Puede usar cualquier CIDR de un espacio de direcciones privado, tal como se define en RFC 1918 Best Current Practice.

Para proporcionar un CIDR personalizado, seleccione Otros en el menú desplegable. Para evitar una colisión de direcciones IP entre OpenSearch Ingestion y Amazon RDS, asegúrese de que el CIDR de la VPC de Amazon Aurora sea diferente del CIDR de OpenSearch Ingestion.

Para obtener más información, consulte Configuring VPC access for a pipeline.

Coherencia de datos

La canalización garantiza la coherencia de datos mediante el sondeo continuo o la recepción de cambios de la instancia de Amazon RDS y la actualización de los documentos correspondientes en el índice de OpenSearch.

OpenSearch Ingestion admite el reconocimiento integral para garantizar la durabilidad de los datos. Cuando una canalización lee instantáneas o transmisiones, crea particiones de forma dinámica para el procesamiento paralelo. La canalización marca una partición como completa cuando recibe un reconocimiento después de incorporar todos los registros en el dominio o la colección de OpenSearch. Si desea incorporarlos a una colección de búsquedas de OpenSearch sin servidor, puede generar un ID de documento en la canalización. Si quiere ingerirlos en una colección de series

temporales de OpenSearch sin servidor, tenga en cuenta que la canalización no genera un ID de documento, de modo que debe omitir `document_id: "${getMetadata(\"primary_key\")}"` en la configuración del receptor de la canalización.

Una canalización de OpenSearch Ingestion también asigna las acciones de los eventos entrantes a las correspondientes acciones de indexación masiva para facilitar la incorporación de documentos. Esto mantiene la coherencia de datos, de modo que cada cambio de datos en Amazon RDS se concilia con los cambios de documento correspondientes en OpenSearch.

Asignación de tipos de datos

La canalización de OpenSearch Ingestion asigna los tipos de datos de PostgreSQL a representaciones que son adecuadas para el consumo de los dominios o las colecciones de OpenSearch Service. Si no se ha definido ninguna plantilla de asignación en OpenSearch, OpenSearch determina automáticamente los tipos de campo con una asignación dinámica en función del primer documento enviado. También puede definir de forma explícita los tipos de campo que mejor se adapten a sus necesidades en OpenSearch mediante una plantilla de asignación.

En la siguiente tabla, se muestran los tipos de datos de RDS para PostgreSQL y los tipos de campos de OpenSearch correspondientes. La columna Tipo de campo de OpenSearch predeterminado muestra el tipo de campo correspondiente en OpenSearch si no se ha definido una asignación explícita. En este caso, OpenSearch determina automáticamente los tipos de campo mediante una asignación dinámica. La columna Tipo de campo recomendado de OpenSearch es el tipo de campo que se recomienda especificar de forma explícita en una plantilla de asignación. Estos tipos de campos están más alineados con los tipos de datos de RDS para PostgreSQL y, por lo general, pueden habilitar mejores características de búsqueda disponibles en OpenSearch.

Tipo de datos de RDS para PostgreSQL	Tipo de campo de OpenSearch predeterminado	Tipo de campo de OpenSearch recomendado
smallint	long	short
entero	long	entero
bigint	long	long

Tipo de datos de RDS para PostgreSQL	Tipo de campo de OpenSearch predeterminado	Tipo de campo de OpenSearch recomendado
decimal	text	doble o palabra
numeric[(p, s)]	text	doble o palabra
real	float	float
double precision	float	double
smallserial	long	short
serial	long	entero
bigserial	long	long
money	objeto	objeto
character varying(n)	text	text
varchar(n)	text	text
character (n)	text	text
char(n)	text	text
bpchar (n)	text	text
BPCHAR	text	text
text	text	text

Tipo de datos de RDS para PostgreSQL	Tipo de campo de OpenSearch predeterminado	Tipo de campo de OpenSearch recomendado
enum	text	text
bytea	text	binario
timestamp [(p)] [sin zona horaria]	largo (en milisegundos de época)	date
timestamp [(p)] con zona horaria	largo (en milisegundos de época)	date
date	largo (en milisegundos de época)	date
time [(p)] [sin zona horaria]	largo (en milisegundos de época)	date
time [(p)] con zona horaria	largo (en milisegundos de época)	date
intervalo [campos] [(p)]	texto (formato ISO8601)	text
booleano	booleano	booleano
point	texto (en formato WKT)	geo_shape
línea	texto (en formato WKT)	geo_shape

Tipo de datos de RDS para PostgreSQL	Tipo de campo de OpenSearch predeterminado	Tipo de campo de OpenSearch recomendado
LSEG	texto (en formato WKT)	geo_shape
BOX	texto (en formato WKT)	geo_shape
path	texto (en formato WKT)	geo_shape
polígono	texto (en formato WKT)	geo_shape
CIRCLE	objeto	objeto
cidr	text	text
inet	text	text
macaddr	text	text
macaddr8	text	text
bit(n)	long	byte, corto, entero o largo (según el número de bits)
bit varying(n)	long	byte, corto, entero o largo (según el número de bits)
json	objeto	objeto
jsonb	objeto	objeto
jsonpath	text	text

Le recomendamos configurar la cola de mensajes fallidos (DLQ) en su canalización de OpenSearch Ingestion. Si ha configurado la cola, OpenSearch Service envía todos los documentos con errores que no se pueden incorporar debido a errores de asignación dinámica.

En caso de que las asignaciones automáticos fallen, puede usar `template_type` y `template_content` en su configuración de canalización para definir reglas de asignación explícitas. Como alternativa, puede crear plantillas de asignación directamente en su dominio o colección de búsqueda antes de iniciar la canalización.

Limitaciones

Tenga en cuenta las siguientes limitaciones cuando configure una canalización de OpenSearch Ingestion para RDS para PostgreSQL:

- La integración solo admite una base de datos PostgreSQL por canalización.
- Actualmente, la integración no admite la ingesta de datos entre regiones; la instancia de Amazon RDS y el dominio de OpenSearch deben estar en la misma Región de AWS.
- Actualmente, la integración no admite la ingesta de datos entre cuentas; la instancia de Amazon RDS y la canalización de OpenSearch Ingestion deben estar en la misma Cuenta de AWS.
- Asegúrese de que la instancia de Amazon RDS tenga habilitada la autenticación mediante AWS Secrets Manager, que es el único mecanismo de autenticación admitido.
- La configuración de canalización existente no se puede actualizar para ingerir datos de una base de datos diferente o de una tabla diferente. Para actualizar la base de datos o el nombre de la tabla de una canalización, debe detener la canalización y reiniciarla con una configuración actualizada o crear una canalización nueva.
- Por lo general, las instrucciones en lenguaje de definición de datos (DDL) no son compatibles. La coherencia de datos no se mantendrá si:
  - Se cambian las claves principales (agregar/eliminar/cambiar el nombre).
  - Las tablas se eliminan o se truncan.
  - Se cambian los nombres de las columnas o los tipos de datos.
- Si las tablas de PostgreSQL que se van a sincronizar no tienen definidas las claves principales, no se garantiza la coherencia de datos. Deberá definir la opción `document_id` personalizada en OpenSearch y asimilar correctamente la configuración para poder sincronizar las actualizaciones y eliminaciones con OpenSearch.
- No se admiten los clústeres de bases de datos Multi-AZ de RDS.
- Versiones compatibles: PostgreSQL 16 y versiones posteriores.

Alarmas de CloudWatch recomendadas

Se recomiendan las siguientes métricas de CloudWatch para supervisar el rendimiento de la canalización de ingesta. Estas métricas pueden ayudarlo a identificar la cantidad de datos procesados a partir de exportaciones y flujos, los errores al procesar eventos de transmisión y exportación y el número de documentos escritos en el destino. Puede configurar las alarmas de CloudWatch para que lleven a cabo una acción cuando una de estas métricas supere un valor especificado durante un periodo de tiempo determinado.

Métrica	Descripción
pipeline-name .rds.credentialsChanged	Esta métrica indica la frecuencia con la que se rotan los secretos de AWS.
pipeline-name .rds.executorRefreshErrors	Esta métrica indica errores al actualizar los secretos de AWS.
pipeline-name .rds.exportRecordsTotal	Esta métrica indica el número de registros exportados desde Amazon Aurora.
pipeline-name .rds.exportRecordsProcessed	Esta métrica indica el número de registros procesados por la canalización de OpenSearch Ingestion.
pipeline-name .rds.exportRecordProcessingErrors	Esta métrica indica el número de errores de procesamiento en una canalización de OpenSearch Ingestion al leer los datos de un clúster de Amazon Aurora.
pipeline-name .rds.exportRecordsSuccessTotal	Esta métrica indica el número total de registros de exportación procesados correctamente.
pipeline-name .rds.expo	Esta métrica indica el número total de registros que no se han podido procesar.

Métrica	Descripción
rtRecordsFailedTotal	
pipeline-name .rds.bytesReceived	Esta métrica indica el número total de bytes recibidos por una canalización de OpenSearch Ingestion.
pipeline-name .rds.bytesProcessed	Esta métrica indica el número total de bytes procesados por una canalización de OpenSearch Ingestion.
pipeline-name .rds.streamRecordsSuccessTotal	Esta métrica indica el número de registros del flujo procesados correctamente.
pipeline-name .rds.streamRecordsFailedTotal	Esta métrica indica el número total de registros del flujo que no se han podido procesar.

Uso de una canalización de OpenSearch Ingestion con Amazon S3

Con OpenSearch Ingestion, puede utilizar Amazon S3 como origen o como destino. Cuando utiliza Amazon S3 como origen, envía datos a una canalización de OpenSearch Ingestion. Cuando utiliza Amazon S3 como destino, escribe datos de una canalización de OpenSearch Ingestion en uno o más buckets de S3.

Temas

- Amazon S3 como origen
- Amazon S3 como destino
- Varias cuentas de Amazon S3 como origen

Amazon S3 como origen

Existen dos formas de utilizar Amazon S3 como origen para procesar datos: con el procesamiento S3-SQS y con análisis programados.

Utilice el procesamiento S3-SQS cuando necesite escanear los archivos casi en tiempo real una vez que se hayan escrito en S3. Puede configurar los buckets de Amazon S3 para que generen un evento cada vez que se almacene o modifique un objeto en el bucket. Utilice un escaneo programado único o recurrente para procesar por lotes los datos de un bucket de S3.

Temas

- Requisitos previos
- Paso 1: configurar el rol de canalización
- Paso 2: crear la canalización

Requisitos previos

Para usar Amazon S3 como origen de una canalización de OpenSearch Ingestion para un escaneo programado o un procesamiento S3-SQS, primero cree un bucket de S3.

> **Note**
>
> Si el bucket de S3 utilizado como origen en la canalización de OpenSearch Ingestion se encuentra en otra Cuenta de AWS, también debe habilitar los permisos de lectura entre cuentas en el bucket. Esto permite que la canalización lea y procese los datos. Para habilitar los permisos entre cuentas, consulte Propietario del bucket que concede permisos de bucket entre cuentas en la Guía del usuario de Amazon S3.
> Si sus buckets de S3 están en varias cuentas, utilice una asignación `bucket_owners`. Para ver un ejemplo, consulte Cross-account S3 access en la documentación de OpenSearch.

Para configurar el procesamiento S3-SQS, también debe realizar los siguientes pasos:

1. Crear una cola de Amazon SQS.
2. Habilitar las notificaciones de eventos en el bucket de S3 con la cola SQS como destino.

Paso 1: configurar el rol de canalización

A diferencia de otros complementos de origen que envían datos a una canalización, el complemento de fuente de S3 tiene una arquitectura basada en lectura en la que la canalización extrae datos de la fuente.

Por lo tanto, para que una canalización pueda leer desde S3, debe especificar un rol dentro de la configuración de fuente de S3 de la canalización que tenga acceso tanto al bucket de S3 como a la cola de Amazon SQS. La canalización asumirá este rol para leer los datos de la cola.

> **Note**
>
> El rol que especifique en la configuración de fuente de S3 debe ser el rol de canalización. Por lo tanto, su rol de canalización debe contener dos políticas de permisos independientes: una para escribir en un receptor y otra para extraer de la fuente de S3. Debe usar el mismo `sts_role_arn` en todos los componentes de la canalización.

El siguiente ejemplo de política muestra los permisos necesarios para usar S3 como fuente:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action":[
          "s3:ListBucket",
          "s3:GetBucketLocation",
          "s3:GetObject"
       ],
      "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/*"
    },
    {
       "Effect":"Allow",
       "Action":"s3:ListAllMyBuckets",
       "Resource":"arn:aws:s3:::*"
    },
    {
      "Effect": "Allow",
```

```
      "Action": [
        "sqs:DeleteMessage",
        "sqs:ReceiveMessage",
        "sqs:ChangeMessageVisibility"
      ],
      "Resource": "arn:aws:sqs:us-east-1:111122223333:MyS3EventSqsQueue"
    }
  ]
}
```

Debe adjuntar estos permisos al rol de IAM que especifique en la opción `sts_role_arn` dentro de la configuración del complemento de fuente de S3:

```
version: "2"
source:
  s3:
    ...
    aws:
      ...
processor:
  ...
sink:
  - opensearch:
      ...
```

Paso 2: crear la canalización

Una vez configurados los permisos, puede configurar una canalización de OpenSearch Ingestion en función de su caso de uso de Amazon S3.

Procesamiento S3-SQS

Para configurar el procesamiento S3-SQS, configure su canalización para especificar S3 como origen y configure las notificaciones de Amazon SQS:

```
version: "2"
s3-pipeline:
  source:
    s3:
      notification_type: "sqs"
      codec:
```

```
        newline: null
      sqs:
        queue_url: "https://sqs.us-east-1amazonaws.com/account-id/ingestion-queue"
      compression: "none"
      aws:
        region: "region"
  processor:
  - grok:
      match:
        message:
        - "%{COMMONAPACHELOG}"
  - date:
      destination: "@timestamp"
      from_time_received: true
  sink:
  - opensearch:
      hosts: ["https://search-domain-endpoint.us-east-1es.amazonaws.com"]
      index: "index-name"
      aws:
        region: "region"
```

Si observa un bajo uso de la CPU al procesar archivos pequeños en Amazon S3, considere la posibilidad de aumentar el rendimiento mediante la modificación del valor de la opción `workers`. Para obtener más información, consulte S3 plugin configuration options.

Escaneo programado

Para configurar un escaneo programado, configure su canalización con un cronograma a nivel de escaneo que se aplique a todos los buckets de S3 o a nivel de bucket. Una programación a nivel de bucket o una configuración de intervalos de escaneo siempre sobrescribe una configuración a nivel de escaneo.

Puede configurar los escaneos programados con un escaneo único, que es ideal para la migración de datos, o un escaneo periódico, que es ideal para el procesamiento por lotes.

Para configurar su canalización para que lea desde Amazon S3, utilice los esquemas de Amazon S3 preconfigurados. Puede editar la parte `scan` de la configuración de la canalización para adaptarla a sus necesidades de programación. Para más información, consulte the section called “Uso de esquemas”.

Escaneo único

Un escaneo único programado se ejecuta una sola vez. En su configuración de la canalización, puede usar una `start_time` y `end_time` para especificar cuándo quiere que se escaneen los objetos del bucket. O bien, puede usar un `range` para especificar el intervalo de tiempo en relación con la hora actual en el que desea que se escaneen los objetos del bucket.

Por ejemplo, un rango configurado para PT4H escanea todos los archivos creados en las últimas cuatro horas. Para configurar un escaneo único para que se ejecute por segunda vez, debe detener y reiniciar la canalización. Si no tiene un rango configurado, también debe actualizar las horas de inicio y finalización.

La siguiente configuración establece un escaneo único de todos los buckets y todos los objetos de esos buckets:

```
version: "2"
log-pipeline:
  source:
    s3:
      codec:
        csv:
      compression: "none"
      aws:
        region: "region"
      acknowledgments: true
      scan:
        buckets:
          - bucket:
              name: my-bucket
              filter:
                include_prefix:
                  - Objects1/
                exclude_suffix:
                  - .jpeg
                  - .png
          - bucket:
              name: my-bucket-2
              key_prefix:
                include:
                  - Objects2/
                exclude_suffix:
                  - .jpeg
                  - .png
      delete_s3_objects_on_read: false
```

```
  processor:
    - date:
        destination: "@timestamp"
        from_time_received: true
  sink:
    - opensearch:
        hosts: ["https://search-domain-endpoint.us-east-1es.amazonaws.com"]
        index: "index-name"
        aws:
          region: "region"
        dlq:
          s3:
            bucket: "dlq-bucket"
            region: "us-east-1"
```

La siguiente configuración establece un escaneo único de todos los buckets durante un período de tiempo específico. Esto significa que S3 procesa solo los objetos con tiempos de creación que se encuentran dentro de este período.

```
scan:
  start_time: 2023-01-21T18:00:00.000Z
  end_time: 2023-04-21T18:00:00.000Z
  buckets:
    - bucket:
        name: my-bucket-1
        filter:
          include:
            - Objects1/
          exclude_suffix:
            - .jpeg
            - .png
    - bucket:
        name: my-bucket-2
        filter:
          include:
            - Objects2/
          exclude_suffix:
            - .jpeg
            - .png
```

La siguiente configuración establece un escaneo único tanto a nivel de escaneo como a nivel del bucket. Las horas de inicio y finalización a nivel del bucket anulan las horas de inicio y finalización a nivel de escaneo.

```
scan:
  start_time: 2023-01-21T18:00:00.000Z
  end_time: 2023-04-21T18:00:00.000Z
  buckets:
    - bucket:
        start_time: 2023-01-21T18:00:00.000Z
        end_time: 2023-04-21T18:00:00.000Z
        name: my-bucket-1
        filter:
          include:
            - Objects1/
          exclude_suffix:
            - .jpeg
            - .png
    - bucket:
        start_time: 2023-01-21T18:00:00.000Z
        end_time: 2023-04-21T18:00:00.000Z
        name: my-bucket-2
        filter:
          include:
            - Objects2/
          exclude_suffix:
            - .jpeg
            - .png
```

Al detener una canalización, se elimina cualquier referencia preexistente de los objetos que la canalización analizó antes de la detención. Si se detiene una sola canalización de análisis, volverá a analizar todos los objetos una vez iniciada, incluso si ya se analizaron. Si necesita detener una sola canalización de análisis, se recomienda cambiar el intervalo de tiempo antes de volver a iniciar la canalización.

Si necesita filtrar los objetos por hora de inicio y hora de finalización, la única opción es detener e iniciar la canalización. Si no necesita filtrar por hora de inicio y hora de finalización, puede filtrar los objetos por nombre. No es necesario detener e iniciar la canalización para filtrar por nombre. Para ello, use `include_prefix` y `exclude_suffix`.

Escaneo periódico

Un escaneo periódico programado ejecuta un escaneo de los buckets de S3 especificados a intervalos regulares y programados. Solo puede configurar estos intervalos a nivel de escaneo, ya que no se admiten configuraciones individuales a nivel de bucket.

En la configuración de la canalización, el `interval` especifica la frecuencia del escaneo periódico, que puede ser entre 30 segundos y 365 días. El primero de estos escaneos siempre se produce al crear la canalización. El `count` define el número total de instancias de escaneo.

La siguiente configuración establece un escaneo periódico, con un retraso de 12 horas entre los escaneos:

```
scan:
  scheduling:
    interval: PT12H
    count: 4
  buckets:
    - bucket:
        name: my-bucket-1
        filter:
          include:
            - Objects1/
          exclude_suffix:
            - .jpeg
            - .png
    - bucket:
        name: my-bucket-2
        filter:
          include:
            - Objects2/
          exclude_suffix:
            - .jpeg
            - .png
```

Amazon S3 como destino

Para escribir datos de una canalización de OpenSearch Ingestion en un bucket de S3, utilice el esquema de S3 preconfigurado para crear una canalización con un receptor de S3. Esta canalización dirige datos selectivos a un receptor de OpenSearch y, al mismo tiempo, envía todos los datos para archivarlos en S3. Para más información, consulte the section called “Uso de esquemas”.

Al crear el receptor de S3, puede especificar el formato que prefiera entre una variedad de códecs de receptor. Por ejemplo, si desea escribir datos en formato de columnas, seleccione el códec Parquet

o Avro. Si prefiere un formato basado en filas, seleccione JSON o NDJSON. Para escribir datos en S3 en un esquema específico, también puede definir un esquema insertado dentro de los códecs de receptor usando el formato Avro.

El siguiente ejemplo define un esquema insertado en un receptor de S3:

```
- s3:
  codec:
    parquet:
      schema: >
        {
           "type" : "record",
           "namespace" : "org.vpcFlowLog.examples",
           "name" : "VpcFlowLog",
           "fields" : [
             { "name" : "version", "type" : "string"},
             { "name" : "srcport", "type": "int"},
             { "name" : "dstport", "type": "int"},
             { "name" : "start", "type": "int"},
             { "name" : "end", "type": "int"},
             { "name" : "protocol", "type": "int"},
             { "name" : "packets", "type": "int"},
             { "name" : "bytes", "type": "int"},
             { "name" : "action", "type": "string"},
             { "name" : "logStatus", "type" : "string"}
           ]
         }
```

Cuando defina este esquema, especifique un superconjunto de todas las claves que pueden estar presentes en los distintos tipos de eventos que su canalización envía a un receptor.

Por ejemplo, si en un evento existe la posibilidad de que falte una clave, añada esa clave al esquema con un valor `null`. Las declaraciones de valores nulos permiten que el esquema procese datos no uniformes (algunos eventos tienen estas claves y otros no). Cuando los eventos entrantes tienen estas claves presentes, sus valores se escriben en los receptores.

Esta definición de esquema actúa como un filtro que solo permite enviar las claves definidas a los receptores y elimina las claves indefinidas de los eventos entrantes.

También puede usar `include_keys` y `exclude_keys` en su receptor para filtrar los datos que se dirigen a otros receptores. Estos dos filtros se excluyen mutuamente, por lo que solo puede usar uno a la vez en su esquema. Además, no puede utilizarlos en esquemas definidos por el usuario.

Para crear canalizaciones con dichos filtros, utilice el esquema de filtro de receptor preconfigurado. Para obtener más información, consulte the section called “Uso de esquemas”.

Varias cuentas de Amazon S3 como origen

Puede conceder acceso a todas las cuentas con Amazon S3 para que las canalizaciones de OpenSearch Ingestion puedan acceder a los buckets de S3 de otra cuenta como origen. Para habilitar los accesos entre cuentas, consulte Propietario del bucket que concede permisos de bucket entre cuentas en la Guía del usuario de Amazon S3. Una vez que haya concedido el acceso, asegúrese de que su rol en la canalización cuente con los permisos necesarios.

A continuación, puede crear una canalización con `bucket_owners` para habilitar el acceso entre cuentas a un bucket de Amazon S3 como origen:

```
s3-pipeline:
 source:
  s3:
   notification_type: "sqs"
   codec:
    csv:
     delimiter: ","
     quote_character: "\""
     detect_header: True
   sqs:
    queue_url: "https://sqs.ap-northeast-1.amazonaws.com/401447383613/test-s3-queue"
   bucket_owners:
    my-bucket-01: 123456789012
    my-bucket-02: 999999999999
   compression: "gzip"
```

Uso de una canalización de OpenSearch Ingestion con Amazon Security Lake

Puede usar el complemento fuente de S3 para incorporar datos de Amazon Security Lake a su canalización de OpenSearch Ingestion. Security Lake centraliza automáticamente los datos de seguridad de los entornos de AWS, los entornos en las instalaciones y los proveedores de SaaS en un lago de datos diseñado específicamente. Puede crear una suscripción que replique los datos de Security Lake en su canalización de OpenSearch Ingestion y, a continuación, los grabe en su dominio de OpenSearch Service o en su colección de OpenSearch sin servidor.

Para configurar su canalización para que lea desde Security Lake, utilice el esquema de Security Lake preconfigurado. El esquema incluye una configuración predeterminada para incorporar los archivos Parquet de Open Cybersecurity Schema Framework (OCSF) de Security Lake. Para obtener más información, consulte the section called “Uso de esquemas”.

Temas

- Uso de una canalización de OpenSearch Ingestion con Amazon Security Lake como origen
- Uso de una canalización de OpenSearch Ingestion con Amazon Security Lake como destino

Uso de una canalización de OpenSearch Ingestion con Amazon Security Lake como origen

Puede usar el complemento de origen de Amazon S3 dentro de su canalización de OpenSearch Ingestion para ingerir datos de Amazon Security Lake. Security Lake centraliza automáticamente los datos de seguridad de los entornos de AWS, los sistemas en las instalaciones y los proveedores de SaaS en un lago de datos diseñado específicamente.

Amazon Security Lake tiene los siguientes atributos de metadatos dentro de una canalización:

- `bucket_name`: nombre del bucket de Amazon S3 creado por Security Lake para almacenar datos de seguridad.
- `path_prefix`: el nombre del origen personalizado definido en la política de roles de IAM de Security Lake.
- `region`: la Región de AWS donde se ubica el bucket de S3 del lago de seguridad.
- `accountID`: el ID de Cuenta de AWS en el que está activado Security Lake.
- `sts_role_arn`: el ARN del rol de IAM diseñado para usarse con Security Lake.

Requisitos previos

Antes de crear su canalización de OpenSearch Ingestion, realice los siguientes pasos:

- Habilitar Security Hub.
- Crear un suscriptor en Security Lake.
  - Seleccione las fuentes que quiere incorporar a su canalización.
  - En el caso de las Credenciales de suscriptor, añada el ID de la Cuenta de AWS donde desea crear la canalización. Para el ID externo, especifique `OpenSearchIngestion-{accountid}`.

- Para el Método de acceso a los datos, seleccione S3.
- Para los Detalles de notificación, seleccione Cola de SQS.

Al crear un suscriptor, Security Lake crea automáticamente dos políticas de permisos insertadas: una para S3 y otra para SQS. Las políticas adoptan el siguiente formato: `AmazonSecurityLake-amzn-s3-demo-bucket-S3` y `AmazonSecurityLake-AWSDemo-SQS`. Para permitir que su canalización acceda a las fuentes del suscriptor, debe asociar los permisos necesarios a su rol de canalización.

Configuración del rol de la canalización

Cree una nueva política de permisos en IAM que combine solo los permisos necesarios de las dos políticas que Security Lake creó automáticamente. El siguiente ejemplo de política muestra el privilegio mínimo necesario para que una canalización de OpenSearch Ingestion lea datos de varias fuentes de Security Lake:

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "s3:GetObject"
         ],
         "Resource":[
            "arn:aws:s3:::aws-security-data-lake-us-east-1-abcde/aws/
LAMBDA_EXECUTION/1.0/*",
            "arn:aws:s3:::aws-security-data-lake-us-east-1-abcde/aws/S3_DATA/1.0/
*",
            "arn:aws:s3:::aws-security-data-lake-us-east-1-abcde/aws/
VPC_FLOW/1.0/*",
            "arn:aws:s3:::aws-security-data-lake-us-east-1-abcde/aws/ROUTE53/1.0/
*",
            "arn:aws:s3:::aws-security-data-lake-us-east-1-abcde/aws/
SH_FINDINGS/1.0/*"
         ]
      },
      {
         "Effect":"Allow",
```

```
            "Action":[
               "sqs:ReceiveMessage",
               "sqs:DeleteMessage"
            ],
            "Resource":[
               "arn:aws:sqs:us-east-1:111122223333:AmazonSecurityLake-abcde-Main-
Queue"
            ]
         }
      ]
}
```

> **Important**
>
> Security Lake no administra la política de roles de canalización por usted. Si agrega o elimina fuentes de su suscripción de Security Lake, debe actualizar la política manualmente. Security Lake crea particiones para cada fuente de registro, por lo que debe agregar o eliminar manualmente los permisos en el rol de canalización.

Debe adjuntar estos permisos al rol de IAM que especifique en la opción `sts_role_arn` incluida en la configuración del complemento fuente de S3, en `sqs`.

```
version: "2"
source:
  s3:
    ...
    sqs:
      queue_url: "https://sqs.us-east-1amazonaws.com/account-id/
AmazonSecurityLake-amzn-s3-demo-bucket-Main-Queue"
    aws:
      ...
processor:
  ...
sink:
  - opensearch:
      ...
```

Crear la canalización

Después de agregar los permisos al rol de la canalización, utilice el esquema de Security Lake preconfigurado para crear la canalización. Para obtener más información, consulte the section called “Uso de esquemas”.

Debe especificar la opción `queue_url` en la configuración fuente de `s3`, que es la URL de la cola de Amazon SQS desde la que debe leerse. Para formatear la URL, localice el Punto de conexión de la suscripción en la configuración del suscriptor y cambie `arn:aws:` a `https://`. Por ejemplo, `https://sqs.`*`us-east-1`*`amazonaws.com/`*`account-id`*`/AmazonSecurityLake-`*`AWSDemo`*`-Main-Queue`.

El `sts_role_arn` que especifica en la configuración de fuente de S3 debe ser el ARN del rol de canalización.

Uso de una canalización de OpenSearch Ingestion con Amazon Security Lake como destino

Utilice el complemento del receptor de Amazon S3 en OpenSearch Ingestion para enviar datos desde cualquier origen compatible a Amazon Security Lake. Security Lake recopila y almacena datos de seguridad de AWS entornos en las instalaciones y proveedores de SaaS en un lago de datos dedicado.

Para configurar su canalización para escribir datos de registro en Security Lake, utilice el esquema de registros de tráfico de firewall preconfigurado. El esquema incluye una configuración predeterminada para recuperar registros de seguridad sin procesar u otros datos almacenados en un bucket de Amazon S3, procesar los registros y normalizarlos. A continuación, asigna los datos al marco Open Cybersecurity Schema Framework (OCSF) y envía los datos transformados compatibles con OCSF a Security Lake.

La canalización tiene los siguientes atributos de metadatos:

- `bucket_name`: nombre del bucket de Amazon S3 creado por Security Lake para almacenar datos de seguridad.
- `path_prefix`: el nombre del origen personalizado definido en la política de roles de IAM de Security Lake.
- `region`: la Región de AWS donde se ubica el bucket de S3 del lago de seguridad.
- `accountID`: el ID de Cuenta de AWS en el que está activado Security Lake.

- `sts_role_arn`: el ARN del rol de IAM diseñado para usarse con Security Lake.

Requisitos previos

Antes de crear una canalización para enviar datos a Security Lake, realice los siguientes pasos:

- Habilitar y configurar Amazon Security Lake: configure Amazon Security Lake para centralizar los datos de seguridad de diversos orígenes. Para obtener instrucciones, consulte Habilitar Security Lake mediante la consola.

  Al seleccionar un origen, elija Ingesta de orígenes de AWS específicos y seleccione uno o más orígenes de registros y eventos que desee ingerir.
- Configure los permisos: configure la función de canalización con los permisos necesarios para escribir datos en Security Lake. Para más información, consulte Rol de canalización.

Crear la canalización

Utilice el esquema de Security Lake preconfigurado para crear la canalización. Para obtener más información, consulte Uso de esquemas para crear una canalización.

Uso de una canalización de OpenSearch Ingestion con Fluent Bit

Este ejemplo de Archivo de configuración de Fluent Bit envía los datos de registro de Fluent Bit a una canalización de OpenSearch Ingestion. Para más información sobre la ingesta de datos de registro, consulte Log Analytics en la documentación de Data Prepper.

Tenga en cuenta lo siguiente:

- El valor `host` debe ser el punto de conexión de su canalización. Por ejemplo, *`pipeline-endpoint.us-east-1`*`osis.amazonaws.com`.
- El valor `aws_service` debe ser `osis`.
- El valor `aws_role_arn` es el ARN del rol de IAM de AWS que el cliente debe asumir y utilizar para la autenticación de Signature Version 4.

```
[INPUT]
  name                  tail
  refresh_interval      5
  path                  /var/log/test.log
```

```
  read_from_head        true

[OUTPUT]
  Name http
  Match *
  Host pipeline-endpoint.us-east-1osis.amazonaws.com
  Port 443
  URI /log/ingest
  Format json
  aws_auth true
  aws_region region
  aws_service osis
  aws_role_arn arn:aws:iam::account-id:role/ingestion-role
  Log_Level trace
  tls On
```

A continuación, puede configurar una canalización de OpenSearch Ingestion como la siguiente, que tiene HTTP como origen:

```
version: "2"
unaggregated-log-pipeline:
  source:
    http:
      path: "/log/ingest"
  processor:
    - grok:
        match:
          log:
            - "%{TIMESTAMP_ISO8601:timestamp} %{NOTSPACE:network_node}
 %{NOTSPACE:network_host} %{IPORHOST:source_ip}:%{NUMBER:source_port:int} ->
 %{IPORHOST:destination_ip}:%{NUMBER:destination_port:int} %{GREEDYDATA:details}"
    - grok:
        match:
          details:
            - "'%{NOTSPACE:http_method} %{NOTSPACE:http_uri}' %{NOTSPACE:protocol}"
            - "TLS%{NOTSPACE:tls_version} %{GREEDYDATA:encryption}"
            - "%{NUMBER:status_code:int} %{NUMBER:response_size:int}"
    - delete_entries:
        with_keys: ["details", "log"]

  sink:
    - opensearch:
        hosts: ["https://search-domain-endpoint.us-east-1es.amazonaws.com"]
```

```
        index: "index_name"
        index_type: custom
        bulk_size: 20
        aws:
          region: "region"
```

Uso de una canalización de OpenSearch Ingestion con Fluentd

Fluentd es un ecosistema de recopilación de datos de código abierto que proporciona SDK para diferentes lenguajes y subproyectos, como Fluent Bit. Este ejemplo de Archivo de configuración de Fluentd envía los datos de registro de Fluentd a una canalización de OpenSearch Ingestion. Para más información sobre la ingesta de datos de registro, consulte Log Analytics en la documentación de Data Prepper.

Tenga en cuenta lo siguiente:

- El valor `endpoint` debe ser el punto de conexión de su canalización. Por ejemplo, *`pipeline-endpoint`*`.`*`us-east-1`*`osis.amazonaws.com/apache-log-pipeline/logs`.
- El valor `aws_service` debe ser `osis`.
- El valor `aws_role_arn` es el ARN del rol de IAM de AWS que el cliente debe asumir y utilizar para la autenticación de Signature Version 4.

```
<source>
  @type tail
  path logs/sample.log
  path_key log
  tag apache
  <parse>
    @type none
  </parse>
</source>

<filter apache>
  @type record_transformer
  <record>
    log ${record["message"]}
  </record>
</filter>

<filter apache>
```

```
  @type record_transformer
  remove_keys message
</filter>

<match apache>
  @type http
  endpoint pipeline-endpoint.us-east-1osis.amazonaws.com/apache-log-pipeline/logs
  json_array true

  <auth>
    method aws_sigv4
    aws_service osis
    aws_region region
    aws_role_arn arn:aws:iam::account-id:role/ingestion-role
  </auth>

  <format>
    @type json
  </format>

  <buffer>
    flush_interval 1s
  </buffer>
</match>
```

A continuación, puede configurar una canalización de OpenSearch Ingestion como la siguiente, que tiene HTTP como origen:

```
version: "2"
apache-log-pipeline:
  source:
    http:
      path: "/${pipelineName}/logs"
  processor:
    - grok:
        match:
          log:
            - "%{TIMESTAMP_ISO8601:timestamp} %{NOTSPACE:network_node}
 %{NOTSPACE:network_host} %{IPORHOST:source_ip}:%{NUMBER:source_port:int} ->
 %{IPORHOST:destination_ip}:%{NUMBER:destination_port:int} %{GREEDYDATA:details}"
  sink:
    - opensearch:
        hosts: ["https://search-domain-endpoint.us-east-1es.amazonaws.com"]
```

```
        index: "index_name"
        aws_region: "region"
        aws_sigv4: true
```

Uso de una canalización de OpenSearch Ingestion con machine learning e inferencia en lotes sin conexión

Las canalizaciones de Amazon OpenSearch Ingestion (OSI) admiten el procesamiento de inferencia en lotes fuera de línea con machine learning (ML) para enriquecer de manera eficiente grandes volúmenes de datos a bajo costo. Utilice la inferencia en lotes sin conexión siempre que tenga conjuntos de datos grandes que puedan procesarse de forma asíncrona. La inferencia en lotes sin conexión funciona con los modelos de Amazon Bedrock y SageMaker. Esta característica está disponible en todas las Regiones de AWS compatibles con OpenSearch Ingestion con dominios de OpenSearch Service 2.17 y versiones posteriores.

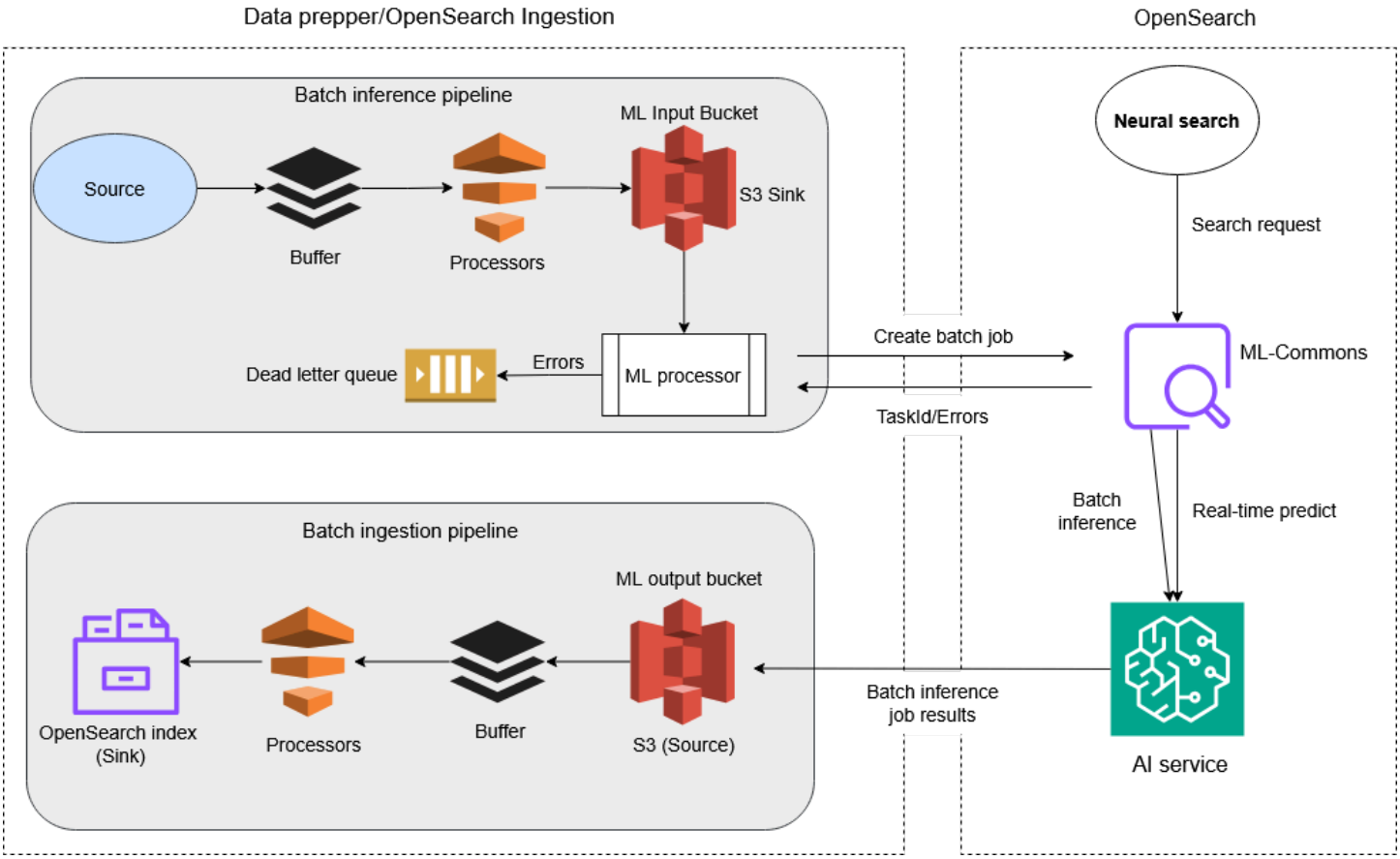
> **Note**
>
> Para el procesamiento de inferencias en tiempo real, utilice Conectores de ML de Amazon OpenSearch Service para plataformas de terceros.

El procesamiento de inferencias en lotes sin conexión aprovecha una característica de OpenSearch llamada ML Commons. ML Commons proporciona algoritmos de ML mediante llamadas a la API de REST y de transporte. Esas llamadas eligen los nodos y los recursos correctos para cada solicitud de ML y supervisan las tareas de ML para garantizar el tiempo de actividad. De esta manera, ML Commons le permite aprovechar los algoritmos de ML de código abierto existentes y reducir el esfuerzo necesario para desarrollar nuevas características de ML. Para obtener más información sobre ML Commons, consulte Machine learning en la documentación de OpenSearch.org.

Funcionamiento

Para crear una canalización de inferencias en lotes sin conexión en OpenSearch Ingestion, puede agregar un procesador de inferencias de machine learning a una canalización. Este procesador permite que su canalización se conecte a servicios de IA como SageMaker para ejecutar tareas de inferencia en lotes. Puede configurar su procesador para que se conecte al servicio de IA que desee a través de los conectores de IA (compatibles con batch_predict) que se ejecutan en su dominio de destino.

OpenSearch Ingestion utiliza el procesador `ml_inference` con ML Commons para crear tareas de inferencia en lotes sin conexión. Luego, ML Commons utiliza la API batch_predict, que realiza inferencias sobre conjuntos de datos de gran tamaño en un modo asíncrono sin conexión mediante un modelo desplegando en servidores de modelos externos en Amazon Bedrock, Amazon SageMaker, Cohere y OpenAI. El siguiente diagrama muestra una canalización de OpenSearch Ingestion que organiza varios componentes para llevar a cabo este proceso de principio a fin:



Los componentes de la canalización funcionan de la siguiente manera:

Canalización 1 (Preparación y transformación de datos)*:

- Origen: los datos se analizan desde un origen externo con la ayuda de OpenSearch Ingestion.
- Procesadores de datos: los datos sin procesar se procesan y transforman al formato correcto para su inferencia en lotes en el servicio de IA integrado.
- S3 (receptor): los datos procesados se almacenan en un bucket de Amazon S3 y están listos para servir como entrada para ejecutar tareas de inferencia en lotes en el servicio de IA integrado.

Canalización 2 (Activar batch_inference de ML):

- Origen: detección automática de eventos de S3 de nuevos archivos creados por la salida de la Canalización 1.
- Procesador ml_inference: procesador que genera inferencias de ML mediante una tarea en lotes asíncrona. Se conecta a los servicios de IA a través del conector de IA configurado que se ejecuta en el dominio de destino.
- ID de tarea: cada tarea en lotes está asociada a un ID de tarea en ml-commons para su seguimiento y administración.
- OpenSearch ML Commons: ML Commons, que aloja el modelo de búsqueda neuronal en tiempo real, administra los conectores a los servidores remotos de IA y proporciona las API para la inferencia en lotes y la administración de tareas.
- Servicios de IA: OpenSearch ML Commons interactúa con servicios de IA como Amazon Bedrock y Amazon SageMaker para realizar inferencias en lotes de los datos y generar predicciones o información. Los resultados se guardan de forma asíncrona en un archivo S3 independiente.

Canalización 3 (Ingesta masiva):

- S3 (origen): los resultados de las tareas en lotes se almacenan en S3, que es el origen de esta canalización.
- Procesadores de transformación de datos: el procesamiento y la transformación adicionales se aplican al resultado de la inferencia en lotes antes de la ingesta. Esto garantiza que los datos estén mapeados correctamente en el índice de OpenSearch.
- Índice de OpenSearch (receptor): los resultados procesados se indexan en OpenSearch para almacenarlos, buscarlos y analizarlos más a fondo.

> **Note**
>
> * El proceso descrito en Canalización 1 es opcional. Si lo prefiere, puede omitir ese proceso y solo cargar los datos preparados en el receptor de S3 para crear tareas en lotes.

Acerca del procesador ml_inference

OpenSearch Ingestion utiliza una integración especializada entre el análisis de S3 y el procesador de inferencias de ML para el procesamiento en lotes. El S3 Scan funciona solo en modo de metadatos para recopilar de manera eficiente la información de los archivos S3 sin leer el contenido real del

archivo. El procesador `ml_inference` utiliza las URL de los archivos de S3 a fin de coordinarlas con ML Commons para el procesamiento en lotes. Este diseño optimiza el flujo de trabajo de inferencia en lotes al minimizar la transferencia de datos innecesaria durante la fase de digitalización. El procesador `ml_inference` se define mediante parámetros. A continuación se muestra un ejemplo:

```
processor:
    - ml_inference:
        # The endpoint URL of your OpenSearch domain
        host: "https://AWStest-offlinebatch-123456789abcdefg.us-
west-2.es.amazonaws.com"

        # Type of inference operation:
        # - batch_predict: for batch processing
        # - predict: for real-time inference
        action_type: "batch_predict"

        # Remote ML model service provider (Amazon Bedrock or SageMaker)
        service_name: "bedrock"

        # Unique identifier for the ML model
        model_id: "AWSTestModelID123456789abcde"

        # S3 path where batch inference results will be stored
        output_path: "s3://amzn-s3-demo-bucket/"

        # Supports ISO_8601 notation strings like PT20.345S or PT15M
        # These settings control how long to keep your inputs in the processor for
 retry on throttling errors
        retry_time_window: "PT9M"

        # AWS configuration settings
        aws:
            # Región de AWS where the Lambda function is deployed
            region: "us-west-2"
            # IAM role ARN for Lambda function execution
            sts_role_arn: "arn:aws::iam::account_id:role/Admin"

        # Dead-letter queue settings for storing errors
        dlq:
          s3:
            region: us-west-2
            bucket: batch-inference-dlq
```

```
            key_path_prefix: bedrock-dlq
            sts_role_arn: arn:aws:iam::account_id:role/OSI-invoke-ml

        # Conditional expression that determines when to trigger the processor
        # In this case, only process when bucket matches "amzn-s3-demo-bucket"
        ml_when: /bucket == "amzn-s3-demo-bucket"
```

Mejoras en el rendimiento de la ingesta mediante el procesador ml_inference

El procesador `ml_inference` de OpenSearch Inestion mejora de forma significativa el rendimiento de la ingesta de datos en las búsquedas habilitadas para ML. El procesador es ideal para casos de uso que requieren datos generados por modelos de machine learning, como la búsqueda semántica, la búsqueda multimodal, el enriquecimiento de documentos y la comprensión de consultas. En la búsqueda semántica, el procesador puede acelerar la creación e ingesta de vectores de gran volumen y alta dimensión en un orden de magnitud.

La capacidad de inferencia en lotes sin conexión del procesador ofrece claras ventajas en comparación con la invocación de modelos en tiempo real. Si bien el procesamiento en tiempo real requiere un servidor modelo en vivo con limitaciones de capacidad, la inferencia en lotes escala de forma dinámica los recursos de cómputo bajo demanda y procesa los datos en paralelo. Por ejemplo, cuando la canalización de OpenSearch Ingestion recibe mil millones de solicitudes de datos de origen, crea 100 archivos de S3 para la entrada de inferencias en lotes de ML. A continuación, el procesador `ml_inference` inicia una tarea en lotes de SageMaker con 100 instancias `ml.m4.xlarge` de Amazon Elastic Compute Cloud (Amazon EC2) y completa la vectorización de mil millones de solicitudes en 14 horas, una tarea que sería prácticamente imposible de realizar en tiempo real.

Configurar el procesador ml_inference de modo que ingiera las solicitudes de datos para una búsqueda semántica

En los siguientes procedimientos, se explica el proceso de instalación y configuración del procesador `ml_inference` de OpenSearch Ingestion para procesar mil millones de solicitudes de datos con fines de búsqueda semántica mediante un modelo de incrustación de texto.

Temas

- Paso 1: Cree conectores y registre modelos en OpenSearch
- Paso 2: Crear una canalización de OpenSearch Ingestion para la inferencia en lotes de ML sin conexión
- Paso 3: preparar los datos para la ingesta

- Paso 4: Supervisar la tarea de inferencia en lotes
- Paso 5: ejecute la búsqueda

Paso 1: Cree conectores y registre modelos en OpenSearch

Para el siguiente procedimiento, utilice batch_inference_sagemaker_connector_blueprint de ML Commons para crear un conector y un modelo en Amazon SageMaker. Si prefiere utilizar las plantillas de integración CloudFormation de OpenSearch, consulte (Procedimiento alternativo) Paso 1: Cree conectores y modelos mediante una plantilla de integración de CloudFormation más adelante en esta sección.

Para crear conectores y registrar modelos en OpenSearch

1. Cree un modelo ML de Deep Java Library (DJL) en SageMaker para la transformación por lotes. Para ver otros modelos de DJL, consulte semantic_search_with_CFN_template_for_Sagemaker en GitHub:

```
POST https://api.sagemaker.us-east-1.amazonaws.com/CreateModel
{
   "ExecutionRoleArn": "arn:aws:iam::123456789012:role/aos_ml_invoke_sagemaker",
   "ModelName": "DJL-Text-Embedding-Model-imageforjsonlines",
   "PrimaryContainer": {
      "Environment": {
         "SERVING_LOAD_MODELS" : "djl://ai.djl.huggingface.pytorch/sentence-
transformers/all-MiniLM-L6-v2"
      },
      "Image": "763104351884.dkr.ecr.us-east-1.amazonaws.com/djl-inference:0.29.0-
cpu-full"
   }
}
```

2. Cree un conector con `batch_predict` como el nuevo tipo `action` en el campo `actions`:

```
POST /_plugins/_ml/connectors/_create
{
  "name": "DJL Sagemaker Connector: all-MiniLM-L6-v2",
  "version": "1",
  "description": "The connector to sagemaker embedding model all-MiniLM-L6-v2",
  "protocol": "aws_sigv4",
  "credential": {
  "roleArn": "arn:aws:iam::111122223333:role/SageMakerRole"
```

```
},
  "parameters": {
    "region": "us-east-1",
    "service_name": "sagemaker",
    "DataProcessing": {
      "InputFilter": "$.text",
      "JoinSource": "Input",
      "OutputFilter": "$"
    },
    "MaxConcurrentTransforms": 100,
    "ModelName": "DJL-Text-Embedding-Model-imageforjsonlines",
    "TransformInput": {
      "ContentType": "application/json",
      "DataSource": {
        "S3DataSource": {
          "S3DataType": "S3Prefix",
          "S3Uri": "s3://offlinebatch/msmarcotests/"
        }
      },
      "SplitType": "Line"
    },
    "TransformJobName": "djl-batch-transform-1-billion",
    "TransformOutput": {
      "AssembleWith": "Line",
      "Accept": "application/json",
      "S3OutputPath": "s3://offlinebatch/msmarcotestsoutputs/"
    },
    "TransformResources": {
      "InstanceCount": 100,
      "InstanceType": "ml.m4.xlarge"
    },
    "BatchStrategy": "SingleRecord"
  },
  "actions": [
    {
      "action_type": "predict",
      "method": "POST",
      "headers": {
        "content-type": "application/json"
      },
      "url": "https://runtime.sagemaker.us-east-1.amazonaws.com/endpoints/
OpenSearch-sagemaker-060124023703/invocations",
      "request_body": "${parameters.input}",
      "pre_process_function": "connector.pre_process.default.embedding",
```

```
      "post_process_function": "connector.post_process.default.embedding"
    },
    {
      "action_type": "batch_predict",
      "method": "POST",
      "headers": {
        "content-type": "application/json"
      },
      "url": "https://api.sagemaker.us-east-1.amazonaws.com/CreateTransformJob",
      "request_body": """{ "BatchStrategy": "${parameters.BatchStrategy}",
 "ModelName": "${parameters.ModelName}", "DataProcessing" :
 ${parameters.DataProcessing}, "MaxConcurrentTransforms":
 ${parameters.MaxConcurrentTransforms}, "TransformInput":
 ${parameters.TransformInput}, "TransformJobName" :
 "${parameters.TransformJobName}", "TransformOutput" :
 ${parameters.TransformOutput}, "TransformResources" :
 ${parameters.TransformResources}}"""
    },
    {
      "action_type": "batch_predict_status",
      "method": "GET",
      "headers": {
        "content-type": "application/json"
      },
      "url": "https://api.sagemaker.us-east-1.amazonaws.com/DescribeTransformJob",
      "request_body": """{ "TransformJobName" :
 "${parameters.TransformJobName}"}"""
    },
    {
      "action_type": "cancel_batch_predict",
      "method": "POST",
      "headers": {
        "content-type": "application/json"
      },
      "url": "https://api.sagemaker.us-east-1.amazonaws.com/StopTransformJob",
      "request_body": """{ "TransformJobName" :
 "${parameters.TransformJobName}"}"""
    }
  ]
}
```

3. Utilice el ID de conector devuelto para registrar el modelo de SageMaker:

```
POST /_plugins/_ml/models/_register
```

```
{
    "name": "SageMaker model for batch",
    "function_name": "remote",
    "description": "test model",
    "connector_id": "example123456789-abcde"
}
```

4. Invoque el modelo con el tipo de acción `batch_predict`:

```
POST /_plugins/_ml/models/teHr3JABBiEvs-eod7sn/_batch_predict
{
  "parameters": {
    "TransformJobName": "SM-offline-batch-transform"
  }
}
```

La respuesta contiene un identificador de tarea para la tarea en lotes:

```
{
 "task_id": "exampleIDabdcefd_1234567",
 "status": "CREATED"
}
```

5. Compruebe el estado de la tarea en lotes llamando a la API Get Task con el ID de tarea:

```
GET /_plugins/_ml/tasks/exampleIDabdcefd_1234567
```

La respuesta contiene el estado de la tarea:

```
{
  "model_id": "nyWbv5EB_tT1A82ZCu-e",
  "task_type": "BATCH_PREDICTION",
  "function_name": "REMOTE",
  "state": "RUNNING",
  "input_type": "REMOTE",
  "worker_node": [
    "WDZnIMcbTrGtnR4Lq9jPDw"
  ],
  "create_time": 1725496527958,
  "last_update_time": 1725496527958,
  "is_async": false,
  "remote_job": {
```

```
    "TransformResources": {
      "InstanceCount": 1,
      "InstanceType": "ml.c5.xlarge"
    },
    "ModelName": "DJL-Text-Embedding-Model-imageforjsonlines",
    "TransformOutput": {
      "Accept": "application/json",
      "AssembleWith": "Line",
      "KmsKeyId": "",
      "S3OutputPath": "s3://offlinebatch/output"
    },
    "CreationTime": 1725496531.935,
    "TransformInput": {
      "CompressionType": "None",
      "ContentType": "application/json",
      "DataSource": {
        "S3DataSource": {
          "S3DataType": "S3Prefix",
          "S3Uri": "s3://offlinebatch/sagemaker_djl_batch_input.json"
        }
      },
      "SplitType": "Line"
    },
    "TransformJobArn": "arn:aws:sagemaker:us-east-1:111122223333:transform-job/SM-
offline-batch-transform15",
    "TransformJobStatus": "InProgress",
    "BatchStrategy": "SingleRecord",
    "TransformJobName": "SM-offline-batch-transform15",
    "DataProcessing": {
      "InputFilter": "$.content",
      "JoinSource": "Input",
      "OutputFilter": "$"
    }
  }
}
```

(Procedimiento alternativo) Paso 1: Cree conectores y modelos mediante una plantilla de integración de CloudFormation

Si lo prefiere, puede utilizar AWS CloudFormation para crear automáticamente todos los conectores y modelos de Amazon SageMaker necesarios para la inferencia de ML. Este enfoque simplifica la configuración mediante el uso de una plantilla preconfigurada disponible en la consola de

Amazon OpenSearch Service. Para obtener más información, consulte Uso de CloudFormation para configurar la inferencia remota para la búsqueda semántica.

Para implementar una pila de CloudFormation que cree todos los conectores y modelos de SageMaker necesarios

1. Abra la consola de Amazon OpenSearch Service.
2. En el panel de navegación, elija Integraciones.
3. En el campo de búsqueda, introduzca y, a continuación**`SageMaker`**, seleccione Integración con modelos de incrustación de texto a través de Amazon SageMaker.
4. Seleccione Configurar dominio y, a continuación, seleccione Configurar dominio de VPC o Configurar dominio público.
5. Ingrese la información en los campos de la plantilla. En Habilitar la inferencia en lotes sin conexión, elija true para aprovisionar recursos para el procesamiento en lotes sin conexión.
6. Elija Crear para crear la pila CloudFormation.
7. Una vez creada la pila, abra la pestaña Salidas en la CloudFormation consola y busque el connector_id y el model_id. Necesitará estos valores más adelante cuando configure la canalización.

Paso 2: Crear una canalización de OpenSearch Ingestion para la inferencia en lotes de ML sin conexión

Utilice el siguiente ejemplo a fin de crear una canalización de OpenSearch Ingestion para la inferencia en lotes de ML sin conexión. Para obtener más información sobre la creación de una canalización para OpenSearch Ingestion, consulte Creación de canalizaciones de Amazon OpenSearch Ingestion.

Antes de empezar

En el siguiente ejemplo, se especifica un ARN de rol de IAM para el parámetro `sts_role_arn`. Utilice el siguiente procedimiento para comprobar que este rol está asignado al rol de backend que tiene acceso a ml-commons en OpenSearch.

1. Desplácese hasta el complemento de OpenSearch Dashboards para ver su dominio de OpenSearch Service. Puede encontrar el punto de conexión de paneles en el panel del dominio de la consola de OpenSearch Service.
2. En el menú principal, seleccione Seguridad, Roles y seleccione el rol ml_full_access.

3. Seleccione Usuarios asignados, Administrar mapeo.
4. En Roles de backend, escriba el ARN del rol de Lambda que necesita permiso para llamar a su dominio. Este es un ejemplo: arn:aws:iam::*111122223333*:role/*lambda-role*
5. Seleccione Asignar y confirme que el usuario o el rol aparecen en Usuarios asignados.

Ejemplo para crear una canalización de OpenSearch Ingestion destinada la inferencia en lotes de ML sin conexión

```
version: '2'
extension:
  osis_configuration_metadata:
    builder_type: visual
sagemaker-batch-job-pipeline:
  source:
    s3:
      acknowledgments: true
      delete_s3_objects_on_read: false
      scan:
        buckets:
          - bucket:
              name: name
              data_selection: metadata_only
              filter:
                include_prefix:
                  - sagemaker/sagemaker_djl_batch_input
                exclude_suffix:
                  - .manifest
          - bucket:
              name: name
              data_selection: data_only
              filter:
                include_prefix:
                  - sagemaker/output/
        scheduling:
          interval: PT6M
      aws:
        region: name
      default_bucket_owner: account_ID
      codec:
        ndjson:
          include_empty_objects: false
```

```
      compression: none
      workers: '1'
  processor:
    - ml_inference:
        host: "https://search-AWStest-offlinebatch-123456789abcdef.us-
west-2.es.amazonaws.com"
        aws_sigv4: true
        action_type: "batch_predict"
        service_name: "sagemaker"
        model_id: "model_ID"
        output_path: "s3://AWStest-offlinebatch/sagemaker/output"
        aws:
          region: "us-west-2"
          sts_role_arn: "arn:aws:iam::account_ID:role/Admin"
        ml_when: /bucket == "AWStest-offlinebatch"
        dlq:
          s3:
            region: us-west-2
            bucket: batch-inference-dlq
            key_path_prefix: bedrock-dlq
            sts_role_arn: arn:aws:iam::account_ID:role/OSI-invoke-ml
    - copy_values:
        entries:
          - from_key: /text
            to_key: chapter
          - from_key: /SageMakerOutput
            to_key: chapter_embedding
          - delete_entries:
            with_keys:
          - text
          - SageMakerOutput
  sink:
    - opensearch:
        hosts: ["https://search-AWStest-offlinebatch-123456789abcdef.us-
west-2.es.amazonaws.com"]
        aws:
          serverless: false
          region: us-west-2
        routes:
          - ml-ingest-route
        index_type: custom
        index: test-nlp-index
  routes:
```

```
    - ml-ingest-route: /chapter != null and /title != null
```

Paso 3: preparar los datos para la ingesta

A fin de preparar los datos para el procesamiento de inferencias en lotes de ML sin conexión, prepare los datos usted mismo con sus propias herramientas o procesos o utilice OpenSearch Data Prepper. Compruebe que los datos estén organizados en el formato correcto, ya sea mediante una canalización para consumir los datos del origen de datos o con la creación de un conjunto de datos de machine learning.

El siguiente ejemplo usa el conjunto de datos MS MARCO, que incluye una colección de consultas de usuarios reales para tareas de procesamiento del lenguaje natural. El conjunto de datos está estructurado en formato JSONL, donde cada línea representa una solicitud enviada al modelo de incrustación de ML:

```
{"_id": "1185869", "text": ")what was the immediate impact of the Paris Peace Treaties
 of 1947?", "metadata": {"world war 2"}}
{"_id": "1185868", "text": "_________ justice is designed to repair the harm to victim,
 the community and the offender caused by the offender criminal act. question 19
 options:", "metadata": {"law"}}
{"_id": "597651", "text": "what is amber", "metadata": {"nothing"}}
{"_id": "403613", "text": "is autoimmune hepatitis a bile acid synthesis disorder",
 "metadata": {"self immune"}}
...
```

Para realizar una prueba con el conjunto de datos MS MARCO, imagine un escenario en el que cree mil millones de solicitudes de entrada distribuidas en 100 archivos, cada uno con 10 millones de solicitudes. Los archivos se almacenarían en Amazon S3 con el prefijo s3://offlinebatch/sagemaker/sagemaker_djl_batch_input/. El proceso de OpenSearch Ingestion escanearía estos 100 archivos simultáneamente e iniciaría una tarea en lotes de SageMaker con 100 trabajadores para su procesamiento paralelo, lo que permitiría la vectorización y la ingesta eficientes de los mil millones de documentos en OpenSearch.

En los entornos de producción, puede utilizar una canalización de OpenSearch Ingestion para generar archivos de S3 para la entrada de inferencias en lotes. La canalización admite varios orígenes de datos y funciona según un cronograma para transformar de manera continua los datos de origen en archivos de S3. Luego, los servidores de IA procesan automáticamente estos archivos mediante tareas en lotes programadas sin conexión, lo que garantiza el procesamiento y la ingesta continuos de datos.

Paso 4: Supervisar la tarea de inferencia en lotes

Puede supervisar las tareas de inferencia en lotes mediante la consola de SageMaker o la AWS CLI. También puede utilizar la API Get Task para supervisar tareas en lotes:

```
GET /_plugins/_ml/tasks/_search
{
  "query": {
    "bool": {
      "filter": [
        {
          "term": {
            "state": "RUNNING"
          }
        }
      ]
    }
  },
  "_source": ["model_id", "state", "task_type", "create_time", "last_update_time"]
}
```

La API devuelve una lista de tareas en lotes activas:

```
{
  "took": 2,
  "timed_out": false,
  "_shards": {
    "total": 5,
    "successful": 5,
    "skipped": 0,
    "failed": 0
  },
  "hits": {
    "total": {
      "value": 3,
      "relation": "eq"
    },
    "max_score": 0.0,
    "hits": [
      {
        "_index": ".plugins-ml-task",
        "_id": "nyWbv5EB_tT1A82ZCu-e",
        "_score": 0.0,
```

```
        "_source": {
          "model_id": "nyWbv5EB_tT1A82ZCu-e",
          "state": "RUNNING",
          "task_type": "BATCH_PREDICTION",
          "create_time": 1725496527958,
          "last_update_time": 1725496527958
        }
      },
      {
        "_index": ".plugins-ml-task",
        "_id": "miKbv5EB_tT1A82ZCu-f",
        "_score": 0.0,
        "_source": {
          "model_id": "miKbv5EB_tT1A82ZCu-f",
          "state": "RUNNING",
          "task_type": "BATCH_PREDICTION",
          "create_time": 1725496528123,
          "last_update_time": 1725496528123
        }
      },
      {
        "_index": ".plugins-ml-task",
        "_id": "kiLbv5EB_tT1A82ZCu-g",
        "_score": 0.0,
        "_source": {
          "model_id": "kiLbv5EB_tT1A82ZCu-g",
          "state": "RUNNING",
          "task_type": "BATCH_PREDICTION",
          "create_time": 1725496529456,
          "last_update_time": 1725496529456
        }
      }
    ]
  }
}
```

Paso 5: ejecute la búsqueda

Tras supervisar la tarea de inferencia en lotes y confirmar que se ha completado, puede ejecutar varios tipos de búsquedas de IA, incluidas las semánticas, las híbridas, las conversacionales (con RAG), las dispersas neuronales y las multimodales. Para obtener más información sobre las búsquedas de IA compatibles con OpenSearch Service, consulte Búsqueda de IA.

Para buscar vectores sin procesar, utilice el tipo de consulta `knn`, proporcione la matriz `vector` como entrada y especifique el `k` número de resultados devueltos:

```
GET /my-raw-vector-index/_search
{
  "query": {
    "knn": {
      "my_vector": {
        "vector": [0.1, 0.2, 0.3],
        "k": 2
      }
    }
  }
}
```

Para ejecutar una búsqueda basada en la IA, utilice el tipo de consulta `neural`. Especifique la entrada `query_text`, el `model_id` del modelo de incrustación que configuró en la canalización de OpenSearch Ingestion y el número `k` de resultados devueltos. Para excluir las incrustaciones de los resultados de búsqueda, especifique el nombre del campo de incrustación en el parámetro: `_source.excludes`

```
GET /my-ai-search-index/_search
{
  "_source": {
    "excludes": [
      "output_embedding"
    ]
  },
  "query": {
    "neural": {
      "output_embedding": {
        "query_text": "What is AI search?",
        "model_id": "mBGzipQB2gmRjlv_dOoB",
        "k": 2
      }
    }
  }
}
```

Uso de una canalización de OpenSearch Ingestion con OpenTelemetry Collector

Este ejemplo de archivo de configuración de OpenTelemetry exporta los datos de rastreo del Recolector de OpenTelemetry y los envía a una canalización de OpenSearch Ingestion. Para más información sobre la ingesta de datos de rastreo, consulte Trace Analytics en la documentación de Data Prepper.

Tenga en cuenta lo siguiente:

- El valor `endpoint` debe incluir el punto de conexión de la canalización. Por ejemplo, `https://`*`pipeline-endpoint`*`.`*`us-east-1`*`osis.amazonaws.com`.
- El valor `service` debe ser `osis`.
- La opción `compression` de OTLP/HTTP Exporter debe coincidir con la opción `compression` del origen de OpenTelemetry de la canalización.

```
extensions:
  sigv4auth:
    region: "region"
    service: "osis"

receivers:
  jaeger:
    protocols:
      grpc:

exporters:
  otlphttp:
    traces_endpoint: "https://pipeline-endpoint.us-east-1.osis.amazonaws.com/v1/traces"
    auth:
      authenticator: sigv4auth
    compression: none

service:
  extensions: [sigv4auth]
  pipelines:
    traces:
      receivers: [jaeger]
      exporters: [otlphttp]
```

A continuación, puede configurar una canalización de OpenSearch Ingestion como la siguiente, en la que se especifica el complemento de rastreo OTel como origen:

```
version: "2"
otel-trace-pipeline:
  source:
    otel_trace_source:
      path: "/v1/traces"
  processor:
    - trace_peer_forwarder:
  sink:
    - pipeline:
        name: "trace-pipeline"
    - pipeline:
        name: "service-map-pipeline"
trace-pipeline:
  source:
    pipeline:
      name: "otel-trace-pipeline"
  processor:
    - otel_traces:
  sink:
    - opensearch:
        hosts: ["https://search-domain-endpoint.us-east-1.es.amazonaws.com"]
        index_type: trace-analytics-raw
        aws:
          region: "region"
service-map-pipeline:
  source:
    pipeline:
      name: "otel-trace-pipeline"
  processor:
    - service_map:
  sink:
    - opensearch:
        hosts: ["https://search-domain-endpoint.us-east-1.es.amazonaws.com"]
        index_type: trace-analytics-service-map
        aws:
          region: "region"
```

Para ver otro ejemplo de canalización, consulte el esquema preconfigurado de Trace Analytics. Para obtener más información, consulte the section called “Uso de esquemas”.

Uso de una canalización de OpenSearch Ingestion con Kafka

Puede usar el complemento de Kafka para transmitir datos desde clústeres de Kafka autoadministrados a los dominios de Amazon OpenSearch Service y a las colecciones de OpenSearch sin servidor. OpenSearch Ingestion admite conexiones desde clústeres de Kafka configurados con redes públicas o privadas (VPC). En este tema, se describen los requisitos previos y los pasos para configurar una canalización de ingesta, incluida la configuración de los ajustes de red y los métodos de autenticación, como el TLS mutuo (mTLS), SASL/SCRAM o IAM.

Migración de datos desde clústeres públicos de Kafka

Puede usar las canalizaciones de OpenSearch Ingestion para migrar datos desde un clúster de Kafka autoadministrado con una configuración pública, lo que significa que el nombre de DNS del dominio se puede resolver públicamente. Para ello, configure una canalización de OpenSearch Ingestion con un clúster de Kafka autogestionado como origen y OpenSearch Service u OpenSearch sin servidor como destino. Esto procesa los datos de streaming desde un clúster de origen autogestionado a un dominio o colección de destino administrado de AWS.

Requisitos previos

Antes de crear su canalización de OpenSearch Ingestion, realice los siguientes pasos:

1. Cree un clúster de Kafka autogestionado con una configuración de red pública. El clúster debe contener los datos que desea ingerir en OpenSearch Service.
2. Cree un dominio de OpenSearch Service o una colección de OpenSearch sin servidor a la que desee migrar datos. Para obtener más información, consulte the section called “ Creación de dominios de OpenSearch Service” y the section called “Creación de colecciones”.
3. Configure la autenticación en el clúster autogestionado con AWS Secrets Manager. Siga los pasos de Rotate AWS Secrets Manager secrets para activar la rotación de secretos.
4. Adjunte una política basada en recursos al dominio o una política de acceso a datos a la colección. Estas políticas de acceso permiten a OpenSearch Ingestion escribir datos de su clúster autogestionado en su dominio o colección.

   El siguiente ejemplo de política de acceso al dominio permite que el rol de canalización, que creará en el paso siguiente, escriba datos en un dominio. Asegúrese de actualizar el `resource` con su propio ARN.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::444455556666:role/pipeline-role"
      },
      "Action": [
        "es:DescribeDomain",
        "es:ESHttp*"
      ],
      "Resource": [
        "arn:aws:es:us-east-1:111122223333:domain/domain-name"
      ]
    }
  ]
}
```

Para crear un rol de IAM con los permisos correctos para acceder a los datos de escritura de la colección o dominio, consulte the section called “Configuración de roles y usuarios”.

Paso 1: configurar el rol de canalización

Una vez establecidos los requisitos previos de la canalización de Kafka, configure el rol de canalización que quiere usar en la configuración de la canalización y agregue permiso para escribir en un dominio de OpenSearch Service o en una colección de OpenSearch sin servidor, así como permiso para leer los secretos de Secrets Manager.

Paso 2: crear la canalización

A continuación, puede configurar una canalización de OpenSearch Ingestion como la siguiente, que especifica Kafka como origen.

Puede especificar varios dominios de OpenSearch Service como destinos para sus datos. Esta capacidad permite el enrutamiento condicional o la replicación de datos entrantes en varios dominios de OpenSearch Service.

También puede migrar datos de un clúster de Confluent Kafka a una colección de VPC de OpenSearch sin servidor. Asegúrese de proporcionar una política de acceso a la red en la configuración de la canalización. Puede utilizar un registro de esquemas de Confluent para definir un esquema de Confluent.

```
version: "2"
kafka-pipeline:
  source:
    kafka:
      encryption:
        type: "ssl"
      topics:
        - name: "topic-name"
          group_id: "group-id"
      bootstrap_servers:
        - "bootstrap-server.us-east-1.aws.private.confluent.cloud:9092"
      authentication:
        sasl:
          plain:
            username: ${aws_secrets:confluent-kafka-secret:username}
            password: ${aws_secrets:confluent-kafka-secret:password}
      schema:
        type: confluent
        registry_url: https://my-registry.us-east-1.aws.confluent.cloud
        api_key: "${{aws_secrets:schema-secret:schema_registry_api_key}}"
        api_secret: "${{aws_secrets:schema-secret:schema_registry_api_secret}}"
        basic_auth_credentials_source: "USER_INFO"
  sink:
  - opensearch:
      hosts: ["https://search-mydomain.us-east-1.es.amazonaws.com"]
      aws:
          region: "us-east-1"
      index: "confluent-index"
extension:
  aws:
    secrets:
      confluent-kafka-secret:
        secret_id: "my-kafka-secret"
        region: "us-east-1"
      schema-secret:
        secret_id: "my-self-managed-kafka-schema"
        region: "us-east-1"
```

Puede utilizar un esquema preconfigurado para crear esta canalización. Para obtener más información, consulte the section called “Uso de esquemas”.

Migración de datos desde clústeres de Kafka en una VPC

También puede utilizar las canalizaciones de OpenSearch Ingestion para migrar datos desde un clúster de Kafka autogestionado que se ejecuta en una VPC. Para ello, configure una canalización de OpenSearch Ingestion con un clúster de Kafka autogestionado como origen y OpenSearch Service u OpenSearch sin servidor como destino. Esto procesa los datos de streaming desde un clúster de origen autogestionado a un dominio o colección de destino administrado de AWS.

Requisitos previos

Antes de crear su canalización de OpenSearch Ingestion, realice los siguientes pasos:

1. Cree un clúster de Kafka autogestionado con una configuración de red de VPC que contenga los datos que desee incorporar a OpenSearch Service.
2. Cree un dominio de OpenSearch Service o una colección de OpenSearch sin servidor a la que desee migrar datos. Para obtener más información, consulte Creating OpenSearch Service domains y Creating collections.
3. Configure la autenticación en el clúster autogestionado con AWS Secrets Manager. Siga los pasos de Rotate AWS Secrets Manager secrets para activar la rotación de secretos.
4. Obtenga el ID de la VPC que tiene acceso a Kafka autogestionado. Elija el CIDR de la VPC que utilizará OpenSearch Ingestion.

> **Note**
>
> Si utiliza la Consola de administración de AWS para crear la canalización, también debe adjuntar la canalización de OpenSearch Ingestion a la VPC para poder usar Kafka autogestionado. Para ello, busque la sección Configuración de la red, seleccione la casilla Adjuntar a la VPC y elija el CIDR de una de las opciones predeterminadas proporcionadas o seleccione una opción propia. Puede usar cualquier CIDR de un espacio de direcciones privado, tal como se define en RFC 1918 Best Current Practice.
> Para proporcionar un CIDR personalizado, seleccione Otros en el menú desplegable. Para evitar una colisión de direcciones IP entre OpenSearch Ingestion y OpenSearch

autogestionado, asegúrese de que el CIDR de la VPC de OpenSearch autogestionado sea diferente del CIDR de OpenSearch Ingestion.

5. Adjunte una política basada en recursos al dominio o una política de acceso a datos a la colección. Estas políticas de acceso permiten a OpenSearch Ingestion escribir datos de su clúster autogestionado en su dominio o colección.

   El siguiente ejemplo de política de acceso al dominio permite que el rol de canalización, que creará en el paso siguiente, escriba datos en un dominio. Asegúrese de actualizar el `resource` con su propio ARN.

   JSON

   ```
   {
     "Version":"2012-10-17",
     "Statement": [
       {
         "Effect": "Allow",
         "Principal": {
           "AWS": "arn:aws:iam::444455556666:role/pipeline-role"
         },
         "Action": [
           "es:DescribeDomain",
           "es:ESHttp*"
         ],
         "Resource": [
           "arn:aws:es:us-east-1:111122223333:domain/domain-name"
         ]
       }
     ]
   }
   ```

   Para crear un rol de IAM con los permisos correctos para acceder a los datos de escritura de la colección o dominio, consulte the section called “Configuración de roles y usuarios”.

Paso 1: configurar el rol de canalización

Una vez configurados los requisitos previos de la canalización, configure el rol de la canalización que desee usar en la configuración de la canalización y agregue los siguientes permisos al rol:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "SecretsManagerReadAccess",
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue"
            ],
            "Resource": ["arn:aws:secretsmanager:us-
east-1:111122223333:secret:secret-name"]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:AttachNetworkInterface",
                "ec2:CreateNetworkInterface",
                "ec2:CreateNetworkInterfacePermission",
                "ec2:DeleteNetworkInterface",
                "ec2:DeleteNetworkInterfacePermission",
                "ec2:DetachNetworkInterface",
                "ec2:DescribeNetworkInterfaces"
            ],
            "Resource": [
                "arn:aws:ec2:*:*:network-interface/*",
                "arn:aws:ec2:*:*:subnet/*",
                "arn:aws:ec2:*:*:security-group/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeDhcpOptions",
                "ec2:DescribeRouteTables",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeSubnets",
                "ec2:DescribeVpcs",
                "ec2:Describe*"
            ],
            "Resource": "*"
        },
```

```
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateTags"
            ],
            "Resource": "arn:aws:ec2:*:*:network-interface/*",
            "Condition": {
               "StringEquals":
                    {
                        "aws:RequestTag/OSISManaged": "true"
                    }
            }
        }
    ]
}
```

Debe proporcionar los permisos de Amazon EC2 anteriores en el rol de IAM que utilice para crear la canalización de OpenSearch Ingestion, ya que la canalización utiliza estos permisos para crear y eliminar una interfaz de red en la VPC. La canalización solo puede acceder al clúster de Kafka a través de esta interfaz de red.

Paso 2: crear la canalización

A continuación, puede configurar una canalización de OpenSearch Ingestion como la siguiente, que especifica Kafka como origen.

Puede especificar varios dominios de OpenSearch Service como destinos para sus datos. Esta capacidad permite el enrutamiento condicional o la replicación de datos entrantes en varios dominios de OpenSearch Service.

También puede migrar datos de un clúster de Confluent Kafka a una colección de VPC de OpenSearch sin servidor. Asegúrese de proporcionar una política de acceso a la red en la configuración de la canalización. Puede utilizar un registro de esquemas de Confluent para definir un esquema de Confluent.

```
 version: "2"
kafka-pipeline:
  source:
    kafka:
      encryption:
        type: "ssl"
```

```
      topics:
        - name: "topic-name"
          group_id: "group-id"
      bootstrap_servers:
        - "bootstrap-server.us-east-1.aws.private.confluent.cloud:9092"
      authentication:
        sasl:
          plain:
            username: ${aws_secrets:confluent-kafka-secret:username}
            password: ${aws_secrets:confluent-kafka-secret:password}
      schema:
        type: confluent
        registry_url: https://my-registry.us-east-1.aws.confluent.cloud
        api_key: "${{aws_secrets:schema-secret:schema_registry_api_key}}"
        api_secret: "${{aws_secrets:schema-secret:schema_registry_api_secret}}"
        basic_auth_credentials_source: "USER_INFO"
  sink:
  - opensearch:
      hosts: ["https://search-mydomain.us-east-1.es.amazonaws.com"]
      aws:
          region: "us-east-1"
      index: "confluent-index"
extension:
  aws:
    secrets:
      confluent-kafka-secret:
        secret_id: "my-kafka-secret"
        region: "us-east-1"
      schema-secret:
        secret_id: "my-self-managed-kafka-schema"
        region: "us-east-1"
```

Puede utilizar un esquema preconfigurado para crear esta canalización. Para obtener más información, consulte the section called “Uso de esquemas”.

Migración de datos desde clústeres de OpenSearch autoadministrados mediante Amazon OpenSearch Ingestion

Puede utilizar una canalización de Amazon OpenSearch Ingestion con Elasticsearch u OpenSearch autoadministrado para migrar datos a los dominios de Amazon OpenSearch Service y a las

colecciones de OpenSearch sin servidor. OpenSearch Ingestion admite configuraciones de redes públicas y privadas para la migración de datos desde Elasticsearch y OpenSearch autoadministrado.

Migración desde clústeres públicos de OpenSearch

Puede usar las canalizaciones de OpenSearch Ingestion para migrar datos desde un clúster de Elasticsearch u OpenSearch autogestionado con una configuración pública, lo que significa que el nombre de DNS del dominio se puede resolver públicamente. Para ello, configure una canalización de OpenSearch Ingestion con OpenSearch o Elasticsearch como origen y OpenSearch Service u OpenSearch sin servidor como destino. Esto migra de forma eficaz los datos desde un clúster de origen autogestionado a un dominio o colección de destino administrado de AWS.

Requisitos previos

Antes de crear su canalización de OpenSearch Ingestion, realice los siguientes pasos:

1. Cree un clúster de Elasticsearch u OpenSearch autogestionado que contenga los datos que desee migrar y configure un nombre de DNS público. Para obtener instrucciones, consulte Crear un clúster en la documentación de OpenSearch.
2. Cree un dominio de OpenSearch Service o una colección de OpenSearch sin servidor a la que desee migrar datos. Para obtener más información, consulte the section called “ Creación de dominios de OpenSearch Service” y the section called “Creación de colecciones”.
3. Configure la autenticación en el clúster autogestionado con AWS Secrets Manager. Siga los pasos de Rotate AWS Secrets Manager secrets para activar la rotación de secretos.
4. Adjunte una política basada en recursos al dominio o una política de acceso a datos a la colección. Estas políticas de acceso permiten a OpenSearch Ingestion escribir datos de su clúster autogestionado en su dominio o colección.

   El siguiente ejemplo de política de acceso al dominio permite que el rol de canalización, que creará en el paso siguiente, escriba datos en un dominio. Asegúrese de actualizar el `resource` con su propio ARN.

   JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
```

```
      "Principal": {
        "AWS": "arn:aws:iam::444455556666:role/pipeline-role"
      },
      "Action": [
        "es:DescribeDomain",
        "es:ESHttp*"
      ],
      "Resource": [
        "arn:aws:es:us-east-1:111122223333:domain/domain-name"
      ]
    }
  ]
}
```

Para crear un rol de IAM con los permisos correctos para acceder a los datos de escritura de la colección o dominio, consulte the section called “Configuración de roles y usuarios”.

Paso 1: configurar el rol de canalización

Una vez establecidos los requisitos previos de la canalización de OpenSearch, configure el rol de canalización que quiere usar en la configuración de la canalización y agregue permiso para escribir en un dominio de OpenSearch Service o en una colección de OpenSearch sin servidor, así como permiso para leer los secretos de Secrets Manager.

Paso 2: crear la canalización

A continuación, puede configurar una canalización de OpenSearch Ingestion como la siguiente, que especifica OpenSearch como origen.

Puede especificar varios dominios de OpenSearch Service como destinos para sus datos. Esta capacidad permite el enrutamiento condicional o la replicación de datos entrantes en varios dominios de OpenSearch Service.

También puede migrar datos de un clúster de OpenSearch o Elasticsearch a una colección de VPC de OpenSearch sin servidor. Asegúrese de proporcionar una política de acceso a la red en la configuración de la canalización.

```
version: "2"
opensearch-migration-pipeline:
  source:
    opensearch:
```

```
      acknowledgments: true
      host: [ "https://my-self-managed-cluster-name:9200" ]
      indices:
        include:
          - index_name_regex: "include-.*"
        exclude:
          - index_name_regex: '\..*'
      authentication:
        username: ${aws_secrets:secret:username}
        password: ${aws_secrets:secret:password}
        scheduling:
           interval: "PT2H"
           index_read_count: 3
           start_time: "2023-06-02T22:01:30.00Z"
  sink:
  - opensearch:
      hosts: ["https://search-mydomain.us-east-1.es.amazonaws.com"]
      aws:
          region: "us-east-1"
          #Uncomment the following lines if your destination is an OpenSearch
 Serverless collection
          #serverless: true
          # serverless_options:
          #     network_policy_name: "network-policy-name"
      index: "${getMetadata(\"opensearch-index\")}"
      document_id: "${getMetadata(\"opensearch-document_id\")}"
      enable_request_compression: true
      dlq:
        s3:
          bucket: "bucket-name"
          key_path_prefix: "apache-log-pipeline/logs/dlq"
          region: "us-east-1"
extension:
  aws:
    secrets:
      secret:
        secret_id: "my-opensearch-secret"
        region: "us-east-1"
        refresh_interval: PT1H
```

Puede utilizar un esquema preconfigurado para crear esta canalización. Para obtener más información, consulte the section called “Uso de esquemas”.

Migración de datos desde clústeres de OpenSearch en una VPC

También puede utilizar las canalizaciones de OpenSearch Ingestion para migrar datos desde un clúster de Elasticsearch u OpenSearch autogestionado que se ejecuta en una VPC. Para ello, configure una canalización de OpenSearch Ingestion con OpenSearch o Elasticsearch autoadministrados como origen y OpenSearch Service u OpenSearch sin servidor como destino. Esto migra de forma eficaz los datos desde un clúster de origen autogestionado a un dominio o colección de destino administrado de AWS.

Requisitos previos

Antes de crear su canalización de OpenSearch Ingestion, realice los siguientes pasos:

1. Cree un clúster de Elasticsearch u OpenSearch autogestionado con una conexión de red de VPC que contenga los datos que desee migrar.
2. Cree un dominio de OpenSearch Service o una colección de OpenSearch sin servidor a la que desee migrar datos. Para obtener más información, consulte Creating OpenSearch Service domains y Creating collections.
3. Configure la autenticación en el clúster autogestionado con AWS Secrets Manager. Siga los pasos de Rotate AWS Secrets Manager secrets para activar la rotación de secretos.
4. Obtenga el ID de la VPC que tiene acceso a Elasticsearch u OpenSearch autogestionado. Elija el CIDR de la VPC que utilizará OpenSearch Ingestion.

> **Note**
>
> Si utiliza la Consola de administración de AWS para crear la canalización, también debe adjuntar la canalización de OpenSearch Ingestion a la VPC para poder usar Elasticsearch u OpenSearch autogestionado. Para ello, busque la sección Opciones de la red de origen, seleccione la casilla Attach to VPC (Adjuntar a la VPC) y elija el CIDR de una de las opciones predeterminadas proporcionadas. Puede usar cualquier CIDR de un espacio de direcciones privado, tal como se define en RFC 1918 Best Current Practice.
> Para proporcionar un CIDR personalizado, seleccione Otros en el menú desplegable.
> Para evitar una colisión de direcciones IP entre OpenSearch Ingestion y OpenSearch autogestionado, asegúrese de que el CIDR de la VPC de OpenSearch autogestionado sea diferente del CIDR de OpenSearch Ingestion.

5. Adjunte una política basada en recursos al dominio o una política de acceso a datos a la colección. Estas políticas de acceso permiten a OpenSearch Ingestion escribir datos de su clúster autogestionado en su dominio o colección.

   El siguiente ejemplo de política de acceso al dominio permite que el rol de canalización, que creará en el paso siguiente, escriba datos en un dominio. Asegúrese de actualizar el `resource` con su propio ARN.

   JSON

   ```
   {
     "Version":"2012-10-17",
     "Statement": [
       {
         "Effect": "Allow",
         "Principal": {
           "AWS": "arn:aws:iam::444455556666:role/pipeline-role"
         },
         "Action": [
           "es:DescribeDomain",
           "es:ESHttp*"
         ],
         "Resource": [
           "arn:aws:es:us-east-1:111122223333:domain/example.com"
         ]
       }
     ]
   }
   ```

   Para crear un rol de IAM con los permisos correctos para acceder a los datos de escritura de la colección o dominio, consulte the section called “Configuración de roles y usuarios”.

Paso 1: configurar el rol de canalización

Una vez configurados los requisitos previos de la canalización, configure el rol de la canalización que desee usar en la configuración de la canalización y agregue los siguientes permisos al rol:

JSON

```
{
```

```
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "SecretsManagerReadAccess",
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue"
            ],
            "Resource": ["arn:aws:secretsmanager:us-
east-1:111122223333:secret:secret-name"]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:AttachNetworkInterface",
                "ec2:CreateNetworkInterface",
                "ec2:CreateNetworkInterfacePermission",
                "ec2:DeleteNetworkInterface",
                "ec2:DeleteNetworkInterfacePermission",
                "ec2:DetachNetworkInterface",
                "ec2:DescribeNetworkInterfaces"
            ],
            "Resource": [
                "arn:aws:ec2:*:*:network-interface/*",
                "arn:aws:ec2:*:*:subnet/*",
                "arn:aws:ec2:*:*:security-group/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeDhcpOptions",
                "ec2:DescribeRouteTables",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeSubnets",
                "ec2:DescribeVpcs",
                "ec2:Describe*"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateTags"
```

```
            ],
            "Resource": "arn:aws:ec2:*:*:network-interface/*",
            "Condition": {
               "StringEquals":
                    {
                        "aws:RequestTag/OSISManaged": "true"
                    }
            }
        }
    ]
}
```

Debe proporcionar los permisos de Amazon EC2 anteriores en el rol de IAM que utilice para crear la canalización de OpenSearch Ingestion, ya que la canalización utiliza estos permisos para crear y eliminar una interfaz de red en la VPC. La canalización solo puede acceder al clúster de OpenSearch a través de esta interfaz de red.

Paso 2: crear la canalización

A continuación, puede configurar una canalización de OpenSearch Ingestion como la siguiente, que especifica OpenSearch como origen.

Puede especificar varios dominios de OpenSearch Service como destinos para sus datos. Esta capacidad permite el enrutamiento condicional o la replicación de datos entrantes en varios dominios de OpenSearch Service.

También puede migrar datos de un clúster de OpenSearch o Elasticsearch a una colección de VPC de OpenSearch sin servidor. Asegúrese de proporcionar una política de acceso a la red en la configuración de la canalización.

```
version: "2"
opensearch-migration-pipeline:
  source:
    opensearch:
      acknowledgments: true
      host: [ "https://my-self-managed-cluster-name:9200" ]
      indices:
        include:
          - index_name_regex: "include-.*"
        exclude:
          - index_name_regex: '\..*'
```

```
      authentication:
        username: ${aws_secrets:secret:username}
        password: ${aws_secrets:secret:password}
        scheduling:
           interval: "PT2H"
           index_read_count: 3
           start_time: "2023-06-02T22:01:30.00Z"
  sink:
  - opensearch:
      hosts: ["https://search-mydomain.us-east-1.es.amazonaws.com"]
      aws:
          region: "us-east-1"
          #Uncomment the following lines if your destination is an OpenSearch
 Serverless collection
          #serverless: true
          # serverless_options:
          #     network_policy_name: "network-policy-name"
      index: "${getMetadata(\"opensearch-index\")}"
      document_id: "${getMetadata(\"opensearch-document_id\")}"
      enable_request_compression: true
      dlq:
        s3:
          bucket: "bucket-name"
          key_path_prefix: "apache-log-pipeline/logs/dlq"
          region: "us-east-1"
extension:
  aws:
    secrets:
      secret:
        secret_id: "my-opensearch-secret"
        region: "us-east-1"
        refresh_interval: PT1H
```

Puede utilizar un esquema preconfigurado para crear esta canalización. Para obtener más información, consulte the section called “Uso de esquemas”.

Uso de una canalización de OpenSearch Ingestion con Amazon Kinesis Data Streams

Utilice una canalización de OpenSearch Ingestion con Amazon Kinesis Data Streams para ingerir datos de registros de flujos de varios flujos en los dominios y colecciones de Amazon OpenSearch Service. La canalización de OpenSearch Ingestion incorpora la infraestructura de OpenSearch

Ingestion para proporcionar una forma de alta escala y baja latencia de ingerir de forma continua los registros de Kinesis.

Temas

- Amazon Kinesis Data Streams como origen
- Amazon Kinesis Data Streams como origen

Amazon Kinesis Data Streams como origen

Con el siguiente procedimiento, aprenderá a configurar una canalización de OpenSearch Ingestion que utilice Amazon Kinesis Data Streams como origen de datos. En esta sección, se describen los requisitos previos necesarios, como la creación de un dominio de OpenSearch Service o una colección de OpenSearch sin servidor y se explican los pasos para configurar el rol de canalización y crear la canalización.

Requisitos previos

Para configurar su canalización, necesitará uno o varios Kinesis Data Streams activos. Estas transmisiones deben recibir registros o estar listas para recibir registros de otros orígenes. Para más información, consulte Overview of OpenSearch Ingestion.

Configuración de su canalización

1. Creación de un dominio de OpenSearch Service o una colección de OpenSearch sin servidor

   Para crear un dominio o una colección, consulte Cómo empezar a usar OpenSearch Ingestion.

   Para crear un rol de IAM con los permisos correctos para acceder a los datos de escritura de la colección o dominio, consulte Políticas basadas en recursos.
2. Configure el rol de canalización con permisos

   Configure el rol de canalización que desee usar en la configuración de canalización y agregue los siguientes permisos. Reemplace los *valores de marcador de posición* con su información.

   JSON

```
{
```

```
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "allowReadFromStream",
            "Effect": "Allow",
            "Action": [
                "kinesis:DescribeStream",
                "kinesis:DescribeStreamConsumer",
                "kinesis:DescribeStreamSummary",
                "kinesis:GetRecords",
                "kinesis:GetShardIterator",
                "kinesis:ListShards",
                "kinesis:ListStreams",
                "kinesis:ListStreamConsumers",
                "kinesis:RegisterStreamConsumer",
                "kinesis:SubscribeToShard"
            ],
            "Resource": [
                "arn:aws:kinesis:us-east-1:111122223333:stream/stream-name"
            ]
        }
    ]
}
```

Si el cifrado del lado del servidor está habilitado en las transmisiones, la siguiente política de AWS KMS permite descifrar los registros. Reemplace los *valores de marcador de posición* con su información.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "allowDecryptionOfCustomManagedKey",
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt",
                "kms:GenerateDataKey"
```

```
            ],
            "Resource": "arn:aws:kms:us-east-1:111122223333:key/key-id"
        }
    ]
}
```

Para que una canalización escriba datos en un dominio, el dominio debe tener una política de acceso a nivel de dominio que permita al rol de canalización sts_role_arn acceder a ellos.

El siguiente ejemplo de política de acceso al dominio permite que el rol de canalización que creó en el paso anterior, (`pipeline-role`), escriba datos en el dominio `ingestion-domain`. Reemplace los *valores de marcador de posición* con su información.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::your-account-id:role/pipeline-role"
      },
      "Action": ["es:DescribeDomain", "es:ESHttp*"],
      "Resource": "arn:aws:es:Región de AWS:account-id:domain/domain-name/*"
    }
  ]
}
```

3. Crear la canalización

Para configurar una canalización de OpenSearch Ingestion, especifique Kinesis-data-streams como origen. Puede encontrar un esquema listo para usar en la consola de OpenSearch Ingestion a fin de crear dicha canalización. (Opcional) Para crear la canalización mediante la AWS CLI, puede utilizar un esquema denominado "**AWS-KinesisDataStreamsPipeline**". Reemplace los *valores de marcador de posición* con su información.

```
version: "2"
kinesis-pipeline:
  source:
```

```
   kinesis_data_streams:
     acknowledgments: true
     codec:
       # Based on whether kinesis records are aggregated or not, you could choose
json, newline or ndjson codec for processing the records.
       # JSON codec supports parsing nested CloudWatch Events into individual log
entries that will be written as documents into OpenSearch.
       # json:
         # key_name: "logEvents"
         # These keys contain the metadata sent by CloudWatch Subscription Filters
         # in addition to the individual log events:
         # include_keys: [ 'owner', 'logGroup', 'logStream' ]
       newline:
     streams:
       - stream_name: "stream name"
         # Enable this if ingestion should start from the start of the stream.
         # initial_position: "EARLIEST"
         # checkpoint_interval: "PT5M"
         # Compression will always be gzip for CloudWatch, but will vary for other
sources:
         # compression: "gzip"
       - stream_name: "stream name"
         # Enable this if ingestion should start from the start of the stream.
         # initial_position: "EARLIEST"
         # checkpoint_interval: "PT5M"
         # Compression will always be gzip for CloudWatch, but will vary for other
sources:
         # compression: "gzip"

       # buffer_timeout: "1s"
       # records_to_accumulate: 100
       # Change the consumer strategy to "polling". Default consumer strategy will
use enhanced "fan-out" supported by KDS.
       # consumer_strategy: "polling"
       # if consumer strategy is set to "polling", enable the polling config
below.
       # polling:
         # max_polling_records: 100
         # idle_time_between_reads: "250ms"
     aws:
       # Provide the Role ARN with access to Amazon Kinesis Data Streams. This
role should have a trust relationship with osis-pipelines.amazonaws.com
       sts_role_arn: "arn:aws:iam::111122223333:role/Example-Role"
       # Provide the Región de AWS of the Data Stream.
```

```
        region: "us-east-1"

  sink:
    - opensearch:
        # Provide an Amazon OpenSearch Serverless domain endpoint
        hosts: [ "https://search-mydomain-1a2a3a4a5a6a7a8a9a0a9a8a7a.us-
east-1.es.amazonaws.com" ]
        index: "index_${getMetadata(\"stream_name\")}"
        # Ensure adding unique document id as a combination of the metadata
 attributes available.
        document_id: "${getMetadata(\"partition_key\")}_
${getMetadata(\"sequence_number\")}_${getMetadata(\"sub_sequence_number\")}"
        aws:
          # Provide a Role ARN with access to the domain. This role should have a
 trust relationship with osis-pipelines.amazonaws.com
          sts_role_arn: "arn:aws:iam::111122223333:role/Example-Role"
          # Provide the Región de AWS of the domain.
          region: "us-east-1"
          # Enable the 'serverless' flag if the sink is an Amazon OpenSearch
 Serverless collection
          serverless: false
          # serverless_options:
            # Specify a name here to create or update network policy for the
 serverless collection
            # network_policy_name: "network-policy-name"
        # Enable the 'distribution_version' setting if the OpenSearch Serverless
 domain is of version Elasticsearch 6.x
        # distribution_version: "es6"
        # Enable and switch the 'enable_request_compression' flag if the default
 compression setting is changed in the domain. See https://docs.aws.amazon.com/
opensearch-service/latest/developerguide/gzip.html
        # enable_request_compression: true/false
        # Optional: Enable the S3 DLQ to capture any failed requests in an S3
 bucket. Delete this entire block if you don't want a DLQ.
        dlq:
          s3:
            # Provide an S3 bucket
            bucket: "your-dlq-bucket-name"
            # Provide a key path prefix for the failed requests
            # key_path_prefix: "kinesis-pipeline/logs/dlq"
            # Provide the region of the bucket.
            region: "us-east-1"
            # Provide a Role ARN with access to the bucket. This role should have a
 trust relationship with osis-pipelines.amazonaws.com
```

```
            sts_role_arn: "arn:aws:iam::111122223333:role/Example-Role"
```

Opciones de configuración

Para ver las opciones de configuración de Kinesis, consulte Opciones de configuración en la documentación de OpenSearch.

Atributos de metadatos disponibles

- stream_name: nombre de las transmisiones de datos de Kinesis desde las que se ingirió el registro
- partition_key: clave de partición del registro de Kinesis Data Streams que se está ingiriendo
- sequence_number: número de secuencia del registro de Kinesis Data Streams que se está ingiriendo
- sub_sequence_number: número de subsecuencia del registro de Kinesis Data Streams que se está ingiriendo

4. (Opcional) Configurar las unidades de cómputo (OCU) recomendadas para la canalización de Kinesis Data Streams

   También se puede configurar una canalización de origen de OpenSearch Kinesis Data Streams para ingerir registros de transmisiones de más de una transmisión. Para una ingesta más rápida, le recomendamos que agregue una unidad de cómputo adicional por cada nueva transmisión que se agregue.

Coherencia de datos

OpenSearch Ingestion admite el reconocimiento integral para garantizar la durabilidad de los datos. Cuando la canalización lee los registros de transmisión de Kinesis, distribuye dinámicamente el trabajo de lectura de los registros de transmisión en función de los particiones asociadas a las transmisiones. La canalización comprobará automáticamente los flujos cuando reciba una confirmación tras ingerir todos los registros del dominio o la colección de OpenSearch. Esto evitará el procesamiento duplicado de los registros de las transmisiones.

Para crear el índice en función del nombre del flujo, defina el índice en la sección del receptor de opensearch como "index_${getMetadata(\"stream_name\")}".

Amazon Kinesis Data Streams como origen

Puede conceder acceso a todas las cuentas con Amazon Kinesis Data Streams para que las canalizaciones de OpenSearch Ingestion puedan acceder a Kinesis Data Streams de otra cuenta como origen. Complete los siguientes pasos para habilitar el acceso entre cuentas:

Configuración del acceso entre cuentas

1. Establezca la política de recursos en la cuenta que tiene la transmisión de Kinesis

   Reemplace los *valores de marcador de posición* con su información.

   JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "StreamReadStatementID",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:role/Pipeline-Role"
            },
            "Action": [
                "kinesis:DescribeStreamSummary",
                "kinesis:GetRecords",
                "kinesis:GetShardIterator",
                "kinesis:ListShards"
            ],
            "Resource": "arn:aws:kinesis:us-east-1:444455556666:stream/
stream-name"
        },
        {
            "Sid": "StreamEFOReadStatementID",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:role/Pipeline-Role"
            },
            "Action": [
                "kinesis:DescribeStreamSummary",
                "kinesis:ListShards"
```

```
            ],
            "Resource": "arn:aws:kinesis:us-
east-1:444455556666:stream/stream-name/consumer/consumer-name"
        }
    ]
}
```

2. (Opcional) Configure una política para consumidores y recursos para consumidores

   Este paso es opcional y solo será necesario si planea utilizar la estrategia Enhanced Fanout Consumer para leer los registros de las transmisiones. Para más información, consulte Develop enhanced fan-out consumers with dedicated throughput.

   a. Configurar consumidor

      Para reutilizar un consumidor existente, puede omitir este paso. Para obtener más información, consulte RegisterStreamConsumer en la Referencia de la API de Amazon Kinesis Data Streams.

      En el siguiente comando de la CLI de ejemplo, reemplace los *valores de marcador de posición* con su propia información.

      Example Ejemplo de comando de la CLI:

      ```
      aws kinesis register-stream-consumer \
      --stream-arn "arn:aws:kinesis:Región de AWS:account-id:stream/stream-name" \
      --consumer-name consumer-name
      ```

   b. Configure la política de recursos para el consumidor

      En la siguiente sentencia, reemplace los *valores de marcador de posición* con su propia información.

      JSON

      ```
      {
          "Version":"2012-10-17",
      ```

```
    "Statement": [
        {
            "Sid": "ConsumerEFOReadStatementID",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:role/Pipeline-Role"
            },
            "Action": [
                "kinesis:DescribeStreamConsumer",
                "kinesis:SubscribeToShard"
            ],
            "Resource": "arn:aws:kinesis:us-
east-1:444455556666:stream/stream-1/consumer/consumer-name"
        }
    ]
}
```

3. Configuración de canalización

   Para la ingesta entre cuentas, agregue los siguientes atributos en `kinesis_data_streams` para cada transmisión:

   - `stream_arn`: el ARN de la transmisión que pertenece a la cuenta en la que existe la transmisión.
   - `consumer_arn`: se trata de un atributo opcional y debe especificarse si se opta por la estrategia de consumo de distribución ramificada mejorada de forma predeterminada. Especifique el nombre real del consumidor para este campo. Reemplace los *valores de marcador de posición* con su información.

```
version: "2"
     kinesis-pipeline:
       source:
         kinesis_data_streams:
           acknowledgments: true
           codec:
             newline:
           streams:
             - stream_arn: "arn:aws:kinesis:region:stream-account-id:stream/stream-
name"
```

```
              consumer_arn: "consumer arn"
              # Enable this if ingestion should start from the start of the
 stream.
              # initial_position: "EARLIEST"
              # checkpoint_interval: "PT5M"
            - stream_arn: "arn:aws:kinesis:region:stream-account-id:stream/stream-
name"
              consumer_arn: "consumer arn"
               # initial_position: "EARLIEST"

            # buffer_timeout: "1s"
            # records_to_accumulate: 100
            # Enable the consumer strategy to "polling". Default consumer strategy
 will use enhanced "fan-out" supported by KDS.
            # consumer_strategy: "polling"
            # if consumer strategy is set to "polling", enable the polling config
 below.
            # polling:
              # max_polling_records: 100
              # idle_time_between_reads: "250ms"
          aws:
            # Provide the Role ARN with access to Kinesis. This role should have a
 trust relationship with osis-pipelines.amazonaws.com
            sts_role_arn: "arn:aws:iam::111122223333:role/Example-Role"
            # Provide the Región de AWS of the domain.
            region: "us-east-1"

      sink:
        - opensearch:
            # Provide an OpenSearch Serverless domain endpoint
            hosts: [ "https://search-mydomain-1a2a3a4a5a6a7a8a9a0a9a8a7a.us-
east-1.es.amazonaws.com" ]
            index: "index_${getMetadata(\"stream_name\")}"
            # Mapping for documentid based on partition key, shard sequence number
 and subsequence number metadata attributes
            document_id: "${getMetadata(\"partition_key\")}_
${getMetadata(\"sequence_number\")}_${getMetadata(\"sub_sequence_number\")}"
            aws:
              # Provide a Role ARN with access to the domain. This role should
 have a trust relationship with osis-pipelines.amazonaws.com
              sts_role_arn: "arn:aws:iam::111122223333:role/Example-Role"
              # Provide the Región de AWS of the domain.
              region: "us-east-1"
```

```
                # Enable the 'serverless' flag if the sink is an OpenSearch
 Serverless collection
                serverless: false
                  # serverless_options:
                  # Specify a name here to create or update network policy for the
 serverless collection
                  # network_policy_name: network-policy-name
              # Enable the 'distribution_version' setting if the OpenSearch
 Serverless domain is of version Elasticsearch 6.x
              # distribution_version: "es6"
              # Enable and switch the 'enable_request_compression' flag if
 the default compression setting is changed in the domain. See https://
docs.aws.amazon.com/opensearch-service/latest/developerguide/gzip.html
              # enable_request_compression: true/false
              # Optional: Enable the S3 DLQ to capture any failed requests in an S3
 bucket. Delete this entire block if you don't want a DLQ.
              dlq:
                s3:
                  # Provide an Amazon S3 bucket
                  bucket: "your-dlq-bucket-name"
                  # Provide a key path prefix for the failed requests
                  # key_path_prefix: "alb-access-log-pipeline/logs/dlq"
                  # Provide the Región de AWS of the bucket.
                  region: "us-east-1"
                  # Provide a Role ARN with access to the bucket. This role should
 have a trust relationship with osis-pipelines.amazonaws.com
                  sts_role_arn: "arn:aws:iam::111122223333:role/Example-Role"
```

4. OSI Pipeline Role Kinesis Data Streams

   a. Política de IAM

      Agregue la siguiente política al rol de canalización. Reemplace los *valores de marcador de posición* con su información.

      JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
```

```
            "Effect": "Allow",
            "Action": [
                "kinesis:DescribeStreamConsumer",
                "kinesis:SubscribeToShard"
            ],
            "Resource": [
            "arn:aws:kinesis:us-east-1:111122223333:stream/my-stream"
            ]
        },
        {
            "Sid": "allowReadFromStream",
            "Effect": "Allow",
            "Action": [
                "kinesis:DescribeStream",
                "kinesis:DescribeStreamSummary",
                "kinesis:GetRecords",
                "kinesis:GetShardIterator",
                "kinesis:ListShards",
                "kinesis:ListStreams",
                "kinesis:ListStreamConsumers",
                "kinesis:RegisterStreamConsumer"
            ],
            "Resource": [
                "arn:aws:kinesis:us-east-1:111122223333:stream/my-stream"
            ]
        }
    ]
}
```

b. Política de confianza

Para poder ingerir datos de la cuenta de transmisión, tendrá que establecer una relación de confianza entre la función de ingesta de canalización y la cuenta de transmisión. Agregue lo siguiente al rol de canalización. Reemplace los *valores de marcador de posición* con su información.

JSON

```
{
  "Version":"2012-10-17",
```

```
  "Statement": [{
     "Effect": "Allow",
     "Principal": {
       "AWS": "arn:aws:iam::111122223333:root"
      },
     "Action": "sts:AssumeRole"
  }]
}
```

Pasos a seguir a continuación

Después de exportar los datos a una canalización, puede consultarlos desde el dominio de OpenSearch Service que está configurado como receptor de la canalización. Los siguientes recursos pueden ayudarle a comenzar:

- Observabilidad
- the section called “Análisis de seguimiento”
- the section called “Lenguaje de procesamiento de canalizaciones”

Uso de una canalización de OpenSearch Ingestion con AWS Lambda

Utilice el procesador de AWS Lambda para enriquecer los datos de cualquier origen o destino compatible con OpenSearch Ingestion mediante código personalizado. Con el procesador de Lambda, puede aplicar sus propias transformaciones o enriquecimientos de datos y, a continuación, devolver los eventos procesados a la canalización para su posterior procesamiento. Este procesador permite un procesamiento de datos personalizado y le brinda un control total sobre la forma en que se manipulan los datos antes de que se procesen.

> **Note**
>
> El límite de tamaño de carga útil para un solo evento procesado por un procesador de Lambda es de 5 MB. Además, el procesador de Lambda solo admite respuestas en formato de matriz JSON.

Requisitos previos

Antes de crear una canalización con un procesador de Lambda, cree los siguientes recursos:

- Una función de AWS Lambda que enriquece y transforma los datos de origen. Para obtener instrucciones, consulte Crear la función de Lambda.
- Un dominio de OpenSearch Service o una colección de OpenSearch sin servidor que actúe como receptor de la canalización. Para obtener más información, consulte the section called " Creación de dominios de OpenSearch Service" y the section called "Creación de colecciones".
- Un rol de canalización que incluye permisos para escribir en el dominio o receptor de la colección. Para obtener más información, consulte the section called "Rol de canalización".

  El rol de canalización también necesita una política de permisos adjunta que le permita invocar la función de Lambda especificada en la configuración de canalización. Por ejemplo:

  JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "allowinvokeFunction",
            "Effect": "Allow",
            "Action": [
                "lambda:invokeFunction",
                "lambda:InvokeAsync",
                "lambda:ListFunctions"
            ],
            "Resource": "arn:aws:lambda:us-
east-1:111122223333:function:function-name"

        }
    ]
}
```

Creación de una canalización

Para utilizar AWS Lambda como procesador, configure una canalización de OpenSearch Ingestion y especifique `aws_lambda` como procesador. También puede usar el esquema de enriquecimiento de

AWS Lambda personalizado para crear la canalización. Para obtener más información, consulte the section called “Uso de esquemas”.

El siguiente ejemplo de canalización recibe datos de un origen HTTP, los enriquece con un procesador de datos y el procesador de AWS Lambda ingiere los datos procesados en un dominio de OpenSearch.

```
version: "2"
lambda-processor-pipeline:
  source:
    http:
      path: "/${pipelineName}/logs"
  processor:
      - date:
        destination: "@timestamp"
        from_time_received: true
    - aws_lambda:
        function_name: "my-lambda-function"

        tags_on_failure: ["lambda_failure"]
        batch:
            key_name: "events"
        aws:
          region: region
  sink:
    - opensearch:
        hosts: [ "https://search-mydomain.us-east-1es.amazonaws.com" ]
        index: "table-index"
        aws:
          region: "region"
          serverless: false
```

La siguiente función de AWS Lambda de ejemplo transforma los datos entrantes mediante la adición de un nuevo par clave-valor (`"transformed": "true"`) a cada elemento de la matriz de eventos proporcionada y, a continuación, devuelve la versión modificada.

```
import json

def lambda_handler(event, context):
    input_array = event.get('events', [])
    output = []
    for input in input_array:
```

```
        input["transformed"] = "true";
        output.append(input)

    return output
```

Agrupación en lotes

Las canalizaciones envían eventos por lotes al procesador de Lambda y ajustan dinámicamente el tamaño del lote para garantizar que se mantenga por debajo del límite de 5 MB.

El siguiente es un ejemplo de lote de canalización:

```
batch:
    key_name: "events"

input_arrary = event.get('events', [])
```

> **Note**
>
> Al crear una canalización, asegúrese de que la opción `key_name` de la configuración del procesador de Lambda coincida con la clave de evento del controlador de Lambda.

Filtrado condicional

El filtrado condicional le permite controlar cuándo el procesador de AWS Lambda invoca la función de Lambda en función de condiciones específicas de los datos de eventos. Esto resulta especialmente útil cuando desea procesar de forma selectiva ciertos tipos de eventos e ignorar otros.

El siguiente ejemplo de configuración utiliza el filtrado condicional:

```
processors:
  - aws_lambda:
      function_name: "my-lambda-function"
      aws:
        region: "region"
      lambda_when: "/sourceIp == 10.10.10.10"
```

Migración de datos entre dominios y colecciones mediante Amazon OpenSearch Ingestion

Puede utilizar las canalizaciones de OpenSearch Ingestion para migrar datos entre dominios de Amazon OpenSearch Service o colecciones de VPC de OpenSearch sin servidor. Para ello, se configura una canalización en la que se configura un dominio o colección como origen y otro dominio o colección como receptor. Esto migra eficazmente los datos de un dominio o colección a otro.

Para migrar los datos, debe tener los siguientes recursos:

- Un dominio de OpenSearch Service o una colección de VPC de OpenSearch sin servidor de origen. Este dominio o colección contiene los datos que quiere migrar. Si usa un dominio, debe ejecutar la versión 1.0 o posterior de OpenSearch o la versión 7.4 o posterior de Elasticsearch. El dominio también debe tener una política de acceso que conceda los permisos adecuados a su rol de canalización.
- Un dominio o colección de VPC independiente al que desee migrar sus datos. Este dominio o colección actuará como receptor de la canalización.
- Un rol de la canalización que OpenSearch Ingestion utilizará para leer y escribir en su colección o dominio. Debe incluir el Nombre de recurso de Amazon (ARN) de este rol en la configuración de la canalización. Para obtener más información, consulte los siguientes recursos:
  - the section called “Concesión de acceso a dominios para canalizaciones”
  - the section called “Concesión de acceso a colecciones para canalizaciones”

Temas

- Limitaciones
- OpenSearch Service como origen
- Especificación de varios receptores de dominios de OpenSearch Service
- Migración de datos a una colección de VPC de OpenSearch sin servidor

Limitaciones

Cuando se designan dominios de OpenSearch Service o colecciones de OpenSearch sin servidor como receptores, se aplican las siguientes limitaciones:

- Una canalización no puede escribir en más de un dominio de VPC.

- Solo puede migrar datos hacia o desde las colecciones de OpenSearch sin servidor que utilizan el acceso a la VPC. No se admiten las colecciones públicas.
- No puede especificar una combinación de dominios públicos y VPC en una única configuración de la canalización.
- Puede incluir un máximo de 20 receptores que no sean canalizaciones en una sola configuración de canalización.
- Puede especificar receptores a partir de un máximo de tres Regiones de AWS diferentes en una sola configuración de canalización.
- Si alguno de los receptores está inactivo durante demasiado tiempo o no cuenta con la capacidad suficiente para recibir los datos entrantes, una canalización con varios receptores puede experimentar una reducción en la velocidad de procesamiento con el tiempo.

OpenSearch Service como origen

El dominio o la colección que especifique como origen es desde donde se migran los datos.

Creación de un rol de canalización en IAM

Para crear la canalización de OpenSearch Ingestion, primero debe crear un rol de canalización para conceder el acceso de lectura y escritura entre dominios o colecciones. Para ello, siga estos pasos:

1. Cree una nueva política de permisos en IAM para adjuntarla al rol de la canalización. Asegúrese de conceder los permisos de lectura desde el origen y de escritura en el receptor. Para obtener más información sobre cómo configurar los permisos de canalización de IAM para los dominios de OpenSearch Service, consulte the section called “Concesión de acceso a dominios para canalizaciones” y the section called “Concesión de acceso a colecciones para canalizaciones”.
2. Especifique los siguientes permisos en el rol de la canalización para leer desde el origen:

   JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":"es:ESHttpGet",
         "Resource":[
            "arn:aws:es:us-east-1:111122223333:domain/domain-name/",
```

```
            "arn:aws:es:us-east-1:111122223333:domain/domain-name/_cat/
indices",
            "arn:aws:es:us-east-1:111122223333:domain/domain-name/_search",
            "arn:aws:es:us-east-1:111122223333:domain/domain-name/_search/
scroll",
            "arn:aws:es:us-east-1:111122223333:domain/domain-name/*/_search"
         ]
      },
      {
         "Effect":"Allow",
         "Action":"es:ESHttpPost",
         "Resource":[
            "arn:aws:es:us-east-1:111122223333:domain/domain-name/*/_search/
point_in_time",
            "arn:aws:es:us-east-1:111122223333:domain/domain-name/*/_search/
scroll"
         ]
      },
      {
         "Effect":"Allow",
         "Action":"es:ESHttpDelete",
         "Resource":[
            "arn:aws:es:us-east-1:111122223333:domain/domain-name/_search/
point_in_time",
            "arn:aws:es:us-east-1:111122223333:domain/domain-name/_search/
scroll"
         ]
      }
   ]
}
```

Creación de una canalización

Tras adjuntar la política al rol de la canalización, utilice el esquema de migración AWSOpenSearchDataMigrationPipeline para crear la canalización. Este esquema incluye una configuración predeterminada para migrar datos entre dominios o colecciones de OpenSearch Service. Para obtener más información, consulte the section called “Uso de esquemas”.

> **Note**
>
> OpenSearch Ingestion utiliza la versión y la distribución del dominio de origen para determinar qué mecanismo utilizar para la migración. Algunas versiones admiten la opción `point_in_time`. OpenSearch sin servidor usa la opción `search_after` porque no admite `point_in_time` o `scroll`.

Es posible que se estén creando nuevos índices durante el proceso de migración o que los documentos se estén actualizando mientras la migración está en curso. Por este motivo, es posible que deba escanear uno o varios escaneos de los datos del índice de su dominio para recoger datos nuevos o actualizados.

Especifique el número de escaneos que se van a ejecutar el configurar `index_read_count` y `interval` en la canalización. El siguiente de ejemplo muestra cómo realizar varios escaneos:

```
scheduling:
    interval: "PT2H"
    index_read_count: 3
    start_time: "2023-06-02T22:01:30.00Z"
```

OpenSearch Ingestion utiliza la siguiente configuración para garantizar que los datos se escriban en el mismo índice y conserven el mismo ID de documento:

```
index: "${getMetadata(\"opensearch-index\")}"
document_id: "${getMetadata(\"opensearch-document_id\")}"
```

Especificación de varios receptores de dominios de OpenSearch Service

Puede especificar varios dominios públicos de OpenSearch Service como destinos para sus datos. Puede utilizar esta capacidad para realizar un enrutamiento condicional o replicar los datos entrantes en varios dominios de OpenSearch Service. Puede especificar hasta 10 dominios públicos diferentes de OpenSearch Service como receptores.

En el siguiente ejemplo, los datos entrantes se enrutan condicionalmente a diferentes dominios de Servicio OpenSearch:

```
...
```

```
  route:
    - 2xx_status: "/response >= 200 and /response < 300"
    - 5xx_status: "/response >= 500 and /response < 600"
  sink:
    - opensearch:
        hosts: [ "https://search-response-2xx.region.es.amazonaws.com" ]
        aws:
          region: "us-east-1"
        index: "response-2xx"
        routes:
          - 2xx_status
    - opensearch:
        hosts: [ "https://search-response-5xx.region.es.amazonaws.com" ]
        aws:
          region: "us-east-1"
        index: "response-5xx"
        routes:
          - 5xx_status
```

Migración de datos a una colección de VPC de OpenSearch sin servidor

Puede utilizar OpenSearch Ingestion para migrar datos entre un dominio de OpenSearch Service de origen o una colección de OpenSearch sin servidor a un receptor de colecciones de VPC. Debe proporcionar una política de acceso a la red en la configuración de la canalización. Para obtener más información sobre la ingesta de datos en las colecciones de VPC de OpenSearch sin servidor, consulte the section called “Tutorial: incorporar datos en una colección”.

Migración de datos a una colección de VPC

1. Cree una colección de OpenSearch sin servidor. Para obtener instrucciones, consulte the section called “Tutorial: incorporar datos en una colección”.
2. Cree una política de red para la colección que especifique el acceso a la VPC tanto para el punto de conexión de la colección como para el punto de conexión de Dashboards. Para obtener instrucciones, consulte the section called “Acceso a la red”.
3. Cree el rol de canalización si todavía no tiene uno. Para obtener instrucciones, consulte the section called “Rol de canalización”.
4. Cree la canalización. Para obtener instrucciones, consulte the section called “Uso de esquemas”.

Uso de SDK de AWS para interactuar con Amazon OpenSearch Ingestion

En esta sección, se incluye un ejemplo de cómo utilizar los SDK de AWS para interactuar con Amazon OpenSearch Ingestion. El ejemplo de código muestra cómo crear un dominio y una canalización y, a continuación, incorporar datos a la canalización.

Temas

- Python

Python

En el siguiente script de ejemplo, se utiliza AWS SDK para Python (Boto3) para crear un rol de canalización de IAM, un dominio en el que escribir datos y una canalización a través de la cual se incorporan datos. A continuación, ingrese un archivo de registro de muestra en la canalización mediante la biblioteca HTTP `requests`.

Ejecute los siguientes comandos para asegurarse de que dispone de todas las dependencias necesarias:

```
pip install boto3
pip install botocore
pip install requests
pip install requests-auth-aws-sigv4
```

Dentro del script, reemplace todas las instancias de *`account-id`* por su ID de Cuenta de AWS.

```
import boto3
import botocore
from botocore.config import Config
import requests
from requests_auth_aws_sigv4 import AWSSigV4
import time

# Build the client using the default credential configuration.
# You can use the CLI and run 'aws configure' to set access key, secret
# key, and default region.

opensearch = boto3.client('opensearch', config=my_config)
```

```
iam = boto3.client('iam', config=my_config)
osis = boto3.client('osis', config=my_config)

domainName = 'test-domain'  # The name of the domain
pipelineName = 'test-pipeline' # The name of the pipeline

def createPipelineRole(iam, domainName):
    """Creates the pipeline role"""
    response = iam.create_policy(
        PolicyName='pipeline-policy',
        PolicyDocument=f'{{\"Version\":\"2012-10-17\",\"Statement\":[{{\"Effect\":
\"Allow\",\"Action\":\"es:DescribeDomain\",\"Resource\":\"arn:aws:es:us-east-1:account-
id:domain\/{domainName}\"}},{{\"Effect\":\"Allow\",\"Action\":\"es:ESHttp*\",\"Resource
\":\"arn:aws:es:us-east-1:account-id:domain\/{domainName}\/*\"}}]}}'
    )
    policyarn = response['Policy']['Arn']

    response = iam.create_role(
        RoleName='PipelineRole',
        AssumeRolePolicyDocument='{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect
\":\"Allow\",\"Principal\":{\"Service\":\"osis-pipelines.amazonaws.com\"},\"Action\":
\"sts:AssumeRole\"}]}'
    )
    rolename=response['Role']['RoleName']

    response = iam.attach_role_policy(
        RoleName=rolename,
        PolicyArn=policyarn
    )

    print('Creating pipeline role...')
    time.sleep(10)
    print('Role created: ' + rolename)

def createDomain(opensearch, domainName):
    """Creates a domain to ingest data into"""
    response = opensearch.create_domain(
        DomainName=domainName,
        EngineVersion='OpenSearch_2.3',
        ClusterConfig={
            'InstanceType': 't2.small.search',
            'InstanceCount': 5,
            'DedicatedMasterEnabled': True,
            'DedicatedMasterType': 't2.small.search',
```

```
            'DedicatedMasterCount': 3
        },
        # Many instance types require EBS storage.
        EBSOptions={
            'EBSEnabled': True,
            'VolumeType': 'gp2',
            'VolumeSize': 10
        },
        AccessPolicies=f'{{\"Version\":\"2012-10-17\",\"Statement\":[{{\"Effect
\":\"Allow\",\"Principal\":{{\"AWS\":\"arn:aws:iam::account-id:role\/PipelineRole
\"}},\"Action\":\"es:*\",\"Resource\":\"arn:aws:es:us-east-1:account-id:domain\/
{domainName}\/*\"}}]}}',
        NodeToNodeEncryptionOptions={
            'Enabled': True
        }
    )
    return(response)

def waitForDomainProcessing(opensearch, domainName):
    """Waits for the domain to be active"""
    try:
        response = opensearch.describe_domain(
            DomainName=domainName
        )
        # Every 30 seconds, check whether the domain is processing.
        while 'Endpoint' not in response['DomainStatus']:
            print('Creating domain...')
            time.sleep(60)
            response = opensearch.describe_domain(
                DomainName=domainName)

        # Once we exit the loop, the domain is ready for ingestion.
        endpoint = response['DomainStatus']['Endpoint']
        print('Domain endpoint ready to receive data: ' + endpoint)
        createPipeline(osis, endpoint)

    except botocore.exceptions.ClientError as error:
        if error.response['Error']['Code'] == 'ResourceNotFoundException':
            print('Domain not found.')
        else:
            raise error

def createPipeline(osis, endpoint):
    """Creates a pipeline using the domain and pipeline role"""
```

```
    try:
        definition = f'version: \"2\"\nlog-pipeline:\n  source:\n    http:\n      path:
 \"/${{pipelineName}}/logs\"\n  processor:\n    - date:\n        from_time_received:
 true\n        destination: \"@timestamp\"\n  sink:\n    - opensearch:\n        hosts:
 [ \"https://{endpoint}\" ]\n        index: \"application_logs\"\n        aws:\n
   region: \"us-east-1\"'
        response = osis.create_pipeline(
            PipelineName=pipelineName,
            MinUnits=4,
            MaxUnits=9,
            PipelineConfigurationBody=definition,
            PipelineRoleArn="arn:aws:iam::account-id:role/PipelineRole"
        )

        response = osis.get_pipeline(
                PipelineName=pipelineName
        )

        # Every 30 seconds, check whether the pipeline is active.
        while response['Pipeline']['Status'] == 'CREATING':
            print('Creating pipeline...')
            time.sleep(30)
            response = osis.get_pipeline(
                PipelineName=pipelineName)

        # Once we exit the loop, the pipeline is ready for ingestion.
        ingestionEndpoint = response['Pipeline']['IngestEndpointUrls'][0]
        print('Pipeline ready to ingest data at endpoint: ' + ingestionEndpoint)
        ingestData(ingestionEndpoint)

    except botocore.exceptions.ClientError as error:
        if error.response['Error']['Code'] == 'ResourceAlreadyExistsException':
            print('Pipeline already exists.')
            response = osis.get_pipeline(
                PipelineName=pipelineName
            )
            ingestionEndpoint = response['Pipeline']['IngestEndpointUrls'][0]
            ingestData(ingestionEndpoint)
        else:
            raise error


def ingestData(ingestionEndpoint):
    """Ingests a sample log file into the pipeline"""
```

```
    endpoint = 'https://' + ingestionEndpoint
    r = requests.request('POST', f'{endpoint}/log-pipeline/logs',

 data='[{"time":"2014-08-11T11:40:13+00:00","remote_addr":"122.226.223.69","status":"404","requ
 http://www.k2proxy.com//hello.html HTTP/1.1","http_user_agent":"Mozilla/4.0
 (compatible; WOW64; SLCC2;)"}]',
    auth=AWSSigV4('osis'))
    print('Ingesting sample log file into pipeline')
    print('Response: ' + r.text)

def main():
    createPipelineRole(iam, domainName)
    createDomain(opensearch, domainName)
    waitForDomainProcessing(opensearch, domainName)

if __name__ == "__main__":
    main()
```

Seguridad en Amazon OpenSearch Ingestion

La seguridad en la nube de AWS es la máxima prioridad. Como cliente de AWS, se beneficia de una arquitectura de red y un centro de datos que se han diseñado para satisfacer los requisitos de seguridad de las organizaciones más exigentes.

La seguridad es una responsabilidad compartida entre AWS y el usuario. El modelo de responsabilidad compartida la describe como seguridad de la nube y seguridad en la nube:

- Seguridad de la nube: AWS es responsable de proteger la infraestructura que ejecuta los servicios de AWS en la nube de AWS. AWS también proporciona servicios que puede utilizar de forma segura. Auditores independientes prueban y verifican periódicamente la eficacia de nuestra seguridad en el marco de los programas de conformidad de AWS.
- Seguridad en la nube: su responsabilidad se determina según el servicio de AWSque utiliza. También eres responsable de otros factores, incluida la confidencialidad de los datos, los requisitos de la empresa y la legislación y la normativa aplicables.

Esta documentación lo ayuda a comprender cómo aplicar el modelo de responsabilidad compartida cuando utiliza OpenSearch Ingestion. En los siguientes temas se muestra cómo configurar OpenSearch Ingestion para cumplir sus objetivos de seguridad y conformidad. También puede

aprender a utilizar otros servicios de AWS que lo ayudan a monitorear y proteger los recursos de OpenSearch Ingestion.

Temas

- Configuración del acceso a la VPC para canalizaciones de Amazon OpenSearch Ingestion
- Configuración de canalizaciones de OpenSearch Ingestion entre cuentas
- Identity and Access Management en Amazon OpenSearch Ingestion
- Registro de llamadas a la API de Amazon OpenSearch Ingestion mediante AWS CloudTrail

Configuración del acceso a la VPC para canalizaciones de Amazon OpenSearch Ingestion

Puede acceder a las canalizaciones de Amazon OpenSearch Ingestion con un punto de conexión de VPC de interfaz. Una VPC es una red virtual dedicada para Cuenta de AWS. Esta infraestructura en la nube está aislada lógicamente de otras redes virtuales de la nube de AWS. Al acceder a una canalización a través de un punto de conexión de VPC, se consigue una comunicación segura entre OpenSearch Ingestion y los demás servicios dentro de la VPC sin necesidad de utilizar una puerta de enlace de Internet, un dispositivo NAT o una conexión de VPN. Todo el tráfico permanece seguro dentro de la nube de AWS.

OpenSearch Ingestion establece esta conexión privada mediante la creación de un punto de conexión de interfaz alimentado por AWS PrivateLink. Crearemos una interfaz de red de punto de conexión en cada subred que especifique durante la creación de la canalización. Se trata de interfaces de red administradas por el solicitante que sirven como punto de entrada para el tráfico destinado a la canalización de OpenSearch Ingestion. También puede elegir crear y administrar los puntos de conexión de la interfaz.

El uso de una VPC le permite hacer cumplir el flujo de datos a través de sus canalizaciones de OpenSearch Ingestion dentro de los límites de la VPC, en lugar de hacerlo a través de la Internet pública. Las canalizaciones que no están dentro de una VPC envían y reciben datos a través de puntos de conexión públicos e Internet.

Una canalización con acceso a VPC puede escribir en dominios de OpenSearch Service públicos o de VPC y en colecciones de OpenSearch sin servidor públicas o de VPC.

Temas

- Consideraciones

- Limitaciones
- Requisitos previos
- Configuración del acceso mediante VPC para una canalización
- Puntos de conexión de VPC autogestionados
- Roles vinculados a servicios para el acceso mediante VPC

Consideraciones

Tenga en cuenta lo siguiente cuando configure el acceso mediante VPC para una canalización.

- No es necesario que una canalización esté en la misma VPC que su receptor. Tampoco es necesario establecer una conexión entre las dos VPC. OpenSearch Ingestion se encarga de conectarlas por usted.
- Solo puede especificar una VPC para la canalización.
- A diferencia de las canalizaciones públicas, una canalización de VPC debe estar en la misma Región de AWS que el receptor de dominios o colecciones en el que se escribe.
- Puede elegir implementar una canalización en una, dos o tres subredes de la VPC. Las subredes se distribuyen en las mismas zonas de disponibilidad en las que están implementadas las unidades de cómputo (OCUs) de Ingestion OpenSearch.
- Si solo implementa una canalización en una subred y la zona de disponibilidad deja de funcionar, no podrá incorporar datos. Para garantizar una alta disponibilidad, le recomendamos que configure las canalizaciones con dos o tres subredes.
- La especificación de grupos de seguridad es opcional. Si no proporciona un grupo de seguridad, OpenSearch Ingestion usa el grupo de seguridad predeterminado que se especifica en la VPC.

Limitaciones

Las canalizaciones con acceso a la VPC presentan las siguientes limitaciones.

- No puede cambiar la configuración de red de una canalización después de crearla. Si lanza una canalización dentro de una VPC, no podrá cambiarla posteriormente a un punto de conexión público y viceversa.
- Puede lanzar la canalización dentro de un punto de conexión de VPC de la interfaz o un punto de conexión público, pero no es posible hacer las dos cosas. Se debe elegir una de ellas al crear una canalización.

- Después de aprovisionar una canalización con acceso a la VPC, no se puede mover a otra VPC, y no puede cambiar la configuración de sus subredes y grupos de seguridad.
- Si su canalización escribe en un receptor de dominios o colecciones que usa un acceso de VPC, no podrá volver atrás más adelante y cambiar el receptor (de VPC o público) una vez creada la canalización. Debe eliminar y volver a crear manualmente la canalización con el nuevo receptor. Aún puede cambiar un receptor de público a un receptor con acceso de VPC.
- No puede proporcionar acceso de incorporación entre cuentas a las canalizaciones de VPC.

Requisitos previos

Antes de poder aprovisionar una canalización con acceso a la VPC, debe hacer lo siguiente:

- Cree una VPC

  Para crear una VPC, puede utilizar la consola de Amazon VPC, la CLI de AWS o uno de los SDK de AWS. Para obtener más información, consulte Uso de VPC en la Guía del usuario de Amazon VPC. Si ya tiene una VPC, puede omitir este paso.
- Reservar direcciones IP

  OpenSearch Ingestion coloca una interfaz de red elástica en cada subred que especifique durante la creación de la canalización. Cada interfaz de red tiene asociada una dirección IP. Debe reservar una dirección IP por subred para las interfaces de red.

Configuración del acceso mediante VPC para una canalización

Puede habilitar el acceso mediante VPC para una canalización desde la consola de OpenSearch Service o mediante el AWS CLI.

Consola

El acceso a la VPC se configura durante la creación de la canalización. Bajo Opciones de red de origen, elija Acceso a VPC y configure los siguientes ajustes:

Opción	Descripción
Administración de puntos de conexión	Elija si quiere crear sus puntos de conexión de VPC o dejar que OpenSearch Ingestion los cree.

Opción	Descripción
de VPC	Seleccione el ID de la nube privada virtual (VPC) que desee utilizar. La VPC y la canalización deben estar en la misma Región de AWS.
Subredes	Seleccione una o varias subredes. OpenSearch Service coloca un punto de conexión de VPC e interfaces de red elásticas en las subredes.
Grupos de seguridad	Seleccione uno o más grupos de seguridad de VPC que permitan que la aplicación requerida llegue a la canalización de OpenSearch Ingestion en los puertos (80 o 443) y protocolos (HTTP o HTTPS) que expone la canalizac ión.
Opciones de conexión de VPC	Si su fuente requiere comunicación entre VPC, como Amazon DocumentD B, OpenSearch autoadministrado o Confluent Kafka, OpenSearch Ingestion crea interfaces de red elásticas (ENI) en las subredes que especifique para conectarse a estas fuentes. OpenSearch Ingestion utiliza los ENI en cada zona de disponibilidad para llegar a las fuentes especificadas. La opción Adjuntar a la VPC conecta la VPC del plano de datos de OpenSearch Ingestion a la VPC especificada. Seleccione una reserva de CIDR para que la VPC administrada implemente la interfaz de red.

CLI

Para configurar el acceso a la VPC mediante el AWS CLI, especifique el parámetro `--vpc-options`:

```
aws osis create-pipeline \
  --pipeline-name vpc-pipeline \
  --min-units 4 \
  --max-units 10 \
  --vpc-options
 SecurityGroupIds={sg-12345678,sg-9012345},SubnetIds=subnet-1212234567834asdf \
  --pipeline-configuration-body "file://pipeline-config.yaml"
```

Puntos de conexión de VPC autogestionados

Al crear una canalización, puede utilizar la administración de puntos de conexión para crear una canalización con puntos de conexión autogestionados o puntos de conexión administrados por el servicio. La administración de puntos de conexión es opcional y utiliza de forma predeterminada los puntos de conexión administrados por OpenSearch Ingestion.

Para crear una canalización con un punto de conexión de VPC autogestionado en la Consola de administración de AWS, consulte Creating pipelines with the OpenSearch Service console. Para crear una canalización con un punto de conexión de VPC autogestionado en la AWS CLI, puede usar el parámetro `--vpc-options` en el comando create-pipeline:

```
--vpc-options SubnetIds=subnet-abcdef01234567890,VpcEndpointManagement=CUSTOMER
```

Puede crear un punto de conexión para la canalización al especificar el servicio del punto de conexión. Para encontrar el servicio del punto de conexión, use el comando get-pipeline, que devuelve una respuesta similar a la siguiente:

```
"vpcEndpointService" : "com.amazonaws.osis.us-east-1.pipeline-
id-1234567890abcdef1234567890",
"vpcEndpoints" : [
  {
    "vpcId" : "vpc-1234567890abcdef0",
    "vpcOptions" : {
      "subnetIds" : [ "subnet-abcdef01234567890", "subnet-021345abcdef6789" ],
      "vpcEndpointManagement" : "CUSTOMER"
    }
  }
```

Utilice el `vpcEndpointService` de la respuesta para crear un punto de conexión de VPC con la Consola de administración de AWS o la AWS CLI.

Si utiliza puntos de conexión de VPC autogestionados, debe habilitar los atributos de DNS `enableDnsSupport` y `enableDnsHostnames` en su VPC. Tenga en cuenta que, si tiene una canalización con un punto de conexión autoadministrado que debe detener y reiniciar, debe volver a crear el punto de conexión de VPC en su cuenta.

Roles vinculados a servicios para el acceso mediante VPC

Un rol vinculado a un servicio es un tipo único de rol de IAM; que delega permisos en un servicio para que pueda crear y administrar recursos en nombre del usuario. Si elige un punto de conexión de VPC administrado por el servicio, OpenSearch Ingestion requiere un rol vinculado a un servicio denominado AWSServiceRoleForAmazonOpenSearchIngestionService para acceder a la VPC, crear el punto de conexión de la canalización y situar interfaces de red en una subred de la VPC.

Si elige un punto de conexión de VPC autogestionado, OpenSearch Ingestion requiere un rol vinculado a un servicio denominado AWSServiceRoleForOpensearchIngestionSelfManagedVpce. Para obtener más información sobre estos roles, sus permisos y cómo se eliminan, consulte the section called “Rol de creación de canalizaciones”.

OpenSearch Ingestion crea automáticamente el rol al crear una canalización de incorporación. Para que esta creación automática se realice correctamente, el usuario que cree la primera canalización en una cuenta debe tener permisos para realizar la acción `iam:CreateServiceLinkedRole`. Para obtener más información, consulte Permisos de roles vinculados a servicios en la Guía del usuario de IAM. Puede ver el rol en la consola (IAM) de AWS Identity and Access Management una vez creado.

Configuración de canalizaciones de OpenSearch Ingestion entre cuentas

Para los orígenes basados en push, como HTTP y OTel, Amazon OpenSearch Ingestion le permite compartir canalizaciones a través de Cuentas de AWS desde una nube privada virtual (VPC) hacia un punto de conexión de canalización en otra VPC separada. Los equipos que comparten análisis con otros equipos de su organización utilizan esta característica como medio más ágil de, por ejemplo, compartir los análisis de registros.

En esta sección, se usan los siguientes términos:

- Propietario de la canalización: la cuenta que posee y administra la canalización de OpenSearch Ingestion. Solo una cuenta puede ser propietaria de una canalización.
- Cuenta de conexión: una cuenta que se conecta a una canalización compartida y la utiliza. Se pueden conectar varias cuentas a la misma canalización.

Para configurar las VPC de forma que compartan las canalizaciones de OpenSearch Ingestion entre Cuentas de AWS, lleve a cabo las siguientes tareas, tal y como se describe aquí:

- (Propietario de la canalización) Conceder a las cuentas conectadas acceso a una canalización

- (Cuenta de conexión) Cree una conexión de punto de conexión de canalización para cada VPC que se conecte

Antes de empezar

Antes de configurar las VPC para compartir las canalizaciones de OpenSearch Ingestion entre Cuentas de AWS, complete las siguientes tareas:

Tarea	Detalles
Crear una o más canalizaciones de OpenSearch Ingestion	Establezca el mínimo de unidades de cómputo de OpenSearch (OSU) en 2 o más. Para obtener más información, consulte Creación de canalizaciones de Amazon OpenSearch Ingestion. Para obtener información acerca de cómo actualizar una canalización, consulte Actualización de las canalizaciones de Amazon OpenSearch Ingestion.
Creación de una o más VPC para OpenSearch Ingestion	Para habilitar el uso compartido de canalizaciones entre cuentas, cualquier VPC implicada en la canalización y los puntos de conexión de la canalización debe configurarse con los siguientes valores de DNS: • `enableDnsSupport=true` • `enableDnsHostnames=true` Para obtener más información, consulte Atributos de DNS para su VPC en la Guía del usuario de Amazon VPC.

Conceder a las cuentas conectadas acceso a una canalización

Los procedimientos de esta sección describen cómo utilizar la consola y la AWS CLI de OpenSearch Service para configurar el acceso a las canalizaciones entre cuentas mediante la creación de una

política de recursos. Una política de recursos permite al propietario de una canalización especificar otras cuentas que pueden acceder a una canalización. Una vez creadas, las políticas de canalización existen mientras exista la canalización o hasta que se elimine la política.

> **Note**
>
> Las políticas de recursos no sustituyen a la autorización estándar de OpenSearch Ingestion mediante los permisos de IAM. Las políticas de recursos son un mecanismo de autorización adicional para permitir el acceso a las canalizaciones entre cuentas.

Temas

- Conceder a las cuentas conectadas acceso a una canalización (consola)
- Conceder a las cuentas conectadas acceso a una canalización (CLI)

Conceder a las cuentas conectadas acceso a una canalización (consola)

Utilice el siguiente procedimiento para conceder a las cuentas conectadas acceso a una canalización mediante la consola de Amazon OpenSearch Service.

Para crear una conexión de punto de conexión en la canalización

1. En la consola de Amazon OpenSearch Service, en el panel de navegación, amplíe Ingesta y, a continuación, seleccione Canalizaciones.
2. En la sección Canalizaciones, elija el nombre de la canalización a la que desee conceder acceso a una cuenta conectada.
3. Elija la pestaña de puntos de conexión de VPC.
4. En la sección Entidades principales autorizadas, seleccione Autorizar cuenta.
5. En el campo ID de Cuenta de AWS, escriba el número de 12 dígitos (ID) de la cuenta y, a continuación, seleccione Autorizar.

Conceder a las cuentas conectadas acceso a una canalización (CLI)

Utilice el siguiente procedimiento para conceder acceso a las cuentas conectadas a una canalización mediante la AWS CLI.

Para conceder a las cuentas conectadas acceso a la canalización

1. Actualice a la versión más reciente de la AWS CLI (versión 2.0). Para obtener más información, consulte Instalación o actualización de la versión más reciente de la AWS CLI.
2. Abra la CLI en la cuenta y Región de AWS con la canalización que desea compartir.
3. Ejecute el siguiente comando para crear una política de recursos para la canalización. Esta política otorga el permiso `osis:CreatePipelineEndpoint` a la canalización. La política incluye un parámetro en el que se pueden enumerar los ID Cuenta de AWS que se van a permitir.

> **Note**
>
> En el siguiente comando, debe utilizar la forma abreviada del identificador de cuenta, proporcionando únicamente el identificador de cuenta de doce dígitos. El uso de un ARN no funcionará. También debe proporcionar el nombre de recurso de Amazon (ARN) de la canalización en el parámetro de la CLI para `resource-arn` y en la política JSON en `Resource`, como se muestra.

```
aws --region region osis-cross-account put-resource-policy \
  --resource-arn arn:aws:osis:region:pipeline-owner-account-ID:pipeline/pipeline-name
  --policy 'IAM-policy'
```

Utilice una política como la siguiente para la política de IAM

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
  {
  "Sid": "AllowAccess",
  "Effect": "Allow",
  "Principal": {
  "AWS": [
  "111122223333",
  "444455556666"
```

```
 ]
 },
 "Action":
 "osis:CreatePipelineEndpoint",
 "Resource": "arn:aws:osis:us-east-1:123456789012:pipeline/pipeline-name"
 }
 ]
}
```

Cree una conexión de punto de conexión de canalización para cada VPC que se conecte

Después de que el propietario de la canalización conceda acceso a una canalización en su VPC mediante el procedimiento anterior, un usuario de la cuenta de conexión crea un punto de conexión de la canalización en su VPC. En esta sección, se incluyen procedimientos para crear puntos de conexión mediante la consola de OpenSearch Service y la AWS CLI. Al crear un punto de conexión, OpenSearch Ingestion realiza las siguientes acciones:

- Crea el rol vinculado al servicio AWSServiceRoleForAmazonOpenSearchIngestionService en su cuenta, si aún no existe. Este rol otorga al usuario de la cuenta que se conecta permiso para llamar a la acción de la API CreatePipelineEndpoint.
- Crea el punto de conexión de la canalización.
- Configura el punto de conexión de la canalización para ingerir datos de la canalización compartida en la VPC propietaria de la canalización.

Temas

- Creación de una conexión de punto de conexión para la canalización (consola)
- Creación de una conexión de punto de conexión para la canalización (CLI)

Creación de una conexión de punto de conexión para la canalización (consola)

Utilice el siguiente procedimiento para crear un punto de conexión de canalización mediante la consola de OpenSearch Service.

Para crear una conexión de punto de conexión en la canalización

1. En la consola de Amazon OpenSearch Service, en el panel de navegación, amplíe Ingestion y, a continuación, seleccione Puntos de conexión de VPC.
2. En la página Puntos de conexión de VPC, elija Crear.
3. En la ubicación de la canalización, elija una opción. Si elige Cuenta corriente, seleccione la canalización de la lista. Si elige Cuenta cruzada, especifique el ARN de la canalización en el campo. El propietario de la canalización debe haber concedido el acceso a la canalización, tal y como se describe en Conceder a las cuentas conectadas acceso a una canalización.
4. En la sección de configuración de VPC, para VPC, elija una VPC de la lista.
5. En Subnets (Subredes), elija una subred.
6. En Grupos de seguridad, elija un grupo.
7. Seleccione Crear punto de conexión.

Espere a que se produzca la transición del estado del punto de conexión que creó a `ACTIVE`. Una vez que la canalización esté `ACTIVE`, verá un nuevo campo con el nombre `ingestEndpointUrl`. Utilice este punto de conexión para acceder a la canalización e recopilar datos mediante un cliente como FluentBit. Para obtener más información acerca del uso de FluentBit para ingerir datos, consulte Uso de una canalización de OpenSearch Ingestion con Fluent Bit.

> **Note**
>
> `ingestEndpointUrl` es la misma URL para todas las cuentas conectadas.

Creación de una conexión de punto de conexión para la canalización (CLI)

Utilice el siguiente procedimiento para crear una conexión en el punto de conexión de canalización mediante AWS CLI.

Para crear una conexión de punto de conexión en la canalización

1. Si aún no lo ha hecho, actualice a la versión más reciente de la AWS CLI (versión 2.0). Para obtener más información, consulte Instalación o actualización de la versión más reciente de la AWS CLI.
2. Abra la CLI en la cuenta de conexión en la Región de AWS con la canalización compartida.
3. Ejecute el siguiente comando para crear una canalización en el punto de conexión.

> **Note**
>
> Proporcione al menos una subred y un grupo de seguridad para la VPC de la cuenta conectada. El grupo de seguridad debe incluir el puerto 443 y los clientes de soporte en la VPC de la cuenta que se conecta.

```
aws osis --region region create-pipeline-endpoint \
  --pipeline-arn arn:aws:osis:region:connecting-account-ID:pipeline/shared-pipeline-name
  --vpc-options SecurityGroupIds={sg-security-group-ID-1,sg-security-group-ID-2},SubnetIds=subnet-subnet-ID
```

4. Ejecute el siguiente comando para enumerar los puntos de conexión en la región especificada en el comando anterior:

```
aws osis-cross-account --region region list-pipeline-endpoints
```

Espere a que se produzca la transición del estado del punto de conexión que creó a `ACTIVE`. Una vez que la canalización esté `ACTIVE`, verá un nuevo campo con el nombre `ingestEndpointUrl`. Utilice este punto de conexión para acceder a la canalización e recopilar datos mediante un cliente como FluentBit. Para obtener más información acerca del uso de FluentBit para ingerir datos, consulte Uso de una canalización de OpenSearch Ingestion con Fluent Bit.

> **Note**
>
> `ingestEndpointUrl` es la misma URL para todas las cuentas conectadas.

Eliminar los punto de conexión de la canalización

Si ya no quiere proporcionar acceso a una canalización compartida, puede eliminar un punto de conexión de la canalización mediante uno de los siguientes métodos:

- Elimine el punto de conexión de la canalización (cuenta de conexión).
- Revoque el punto de conexión de la canalización (propietario de la canalización).

Siga este procedimiento para eliminar un punto de conexión de la canalización en una cuenta de conexión.

Para eliminar un punto de conexión de la canalización (cuenta de conexión)

1. Abra la CLI en la cuenta de conexión en la Región de AWS con la canalización compartida.
2. Ejecute el siguiente comando para enumerar un punto de conexión de las canalizaciones en la región:

   ```
   aws osis-cross-account --region region list-pipeline-endpoints
   ```

   Anote el ID de canalización que desea eliminar.
3. Ejecute el siguiente comando para eliminar el punto de conexión de la canalización:

   ```
   aws osis-cross-account --region region delete-pipeline-endpoint \
     --endpoint-id 'ID'
   ```

Como propietario de la canalización compartida, utilice el siguiente procedimiento para revocar un punto de conexión de la canalización.

Para revocar un punto de conexión de la canalización (propietario de la canalización)

1. Abra la CLI en la cuenta de conexión en la Región de AWS con la canalización compartida.
2. Ejecute el siguiente comando para enumerar las conexiones de punto de conexión de las canalizaciones en la región:

   ```
   aws osis-cross-account --region region list-pipeline-endpoint-connections
   ```

   Anote el ID de canalización que desea eliminar.
3. Ejecute el siguiente comando para eliminar el punto de conexión de la canalización:

   ```
   aws osis-cross-account --region region revoke-pipeline-endpoint-connections \
     --pipeline-arn pipeline-arn --endpoint-ids ID
   ```

   El comando permite especificar solo un ID de punto de conexión.

Identity and Access Management en Amazon OpenSearch Ingestion

AWS Identity and Access Management (IAM) es un Servicio de AWS que ayuda a los administradores a controlar de forma segura el acceso a los recursos de AWS. Los administradores de IAM controlan quién puede autenticarse (iniciar sesión) y autorizar (tener permisos) para utilizar los recursos de OpenSearch Ingestion. IAM es un servicio de Servicio de AWS que se puede utilizar sin cargo adicional.

Temas

- Políticas basadas en identidades de OpenSearch Ingestion
- Acciones de políticas de OpenSearch Ingestion
- Recursos de políticas de OpenSearch Ingestion
- Claves de condición de política de Amazon OpenSearch Ingestion
- ABAC con OpenSearch Ingestion
- Uso de credenciales temporales con OpenSearch Ingestion
- Roles vinculados a servicios para OpenSearch Ingestion
- Ejemplos de políticas basadas en identidades para OpenSearch Ingestion

Políticas basadas en identidades de OpenSearch Ingestion

Compatibilidad con las políticas basadas en identidad: sí

Las políticas basadas en identidad son documentos de políticas de permisos JSON que puede asociar a una identidad, como un usuario de IAM, un grupo de usuarios o un rol. Estas políticas controlan qué acciones pueden realizar los usuarios y los roles, en qué recursos y en qué condiciones. Para obtener más información sobre cómo crear una política basada en identidad, consulte Creación de políticas de IAM en la Guía del usuario de IAM.

Con las políticas basadas en identidades de IAM, puede especificar las acciones y los recursos permitidos o denegados, así como las condiciones en las que se permiten o deniegan las acciones. Para obtener más información sobre los elementos que puede utilizar en una política de JSON, consulte Referencia de los elementos de las políticas de JSON de IAM en la Guía del usuario de IAM.

Ejemplos de políticas basadas en identidades para OpenSearch Ingestion

Para ver ejemplos de políticas basadas en identidad de OpenSearch Ingestion, consulte the section called “Ejemplos de políticas basadas en identidades”.

Acciones de políticas de OpenSearch Ingestion

Compatibilidad con las acciones de políticas: sí

El elemento `Action` de una política JSON describe las acciones que puede utilizar para conceder o denegar el acceso en una política. Las acciones de la política generalmente tienen el mismo nombre que la operación de API de AWS asociada. Hay algunas excepciones, como acciones de solo permiso que no tienen una operación de API coincidente. También hay algunas operaciones que requieren varias acciones en una política. Estas acciones adicionales se denominan acciones dependientes.

Incluya acciones en una política para conceder permisos y así llevar a cabo la operación asociada.

Las acciones de políticas de OpenSearch Ingestion utilizan el siguiente prefijo antes de la acción:

```
osis
```

Para especificar varias acciones en una única instrucción, sepárelas con comas.

```
"Action": [
      "osis:action1",
      "osis:action2"
         ]
```

Puede especificar varias acciones utilizando caracteres comodín (*). Por ejemplo, para especificar todas las acciones que comiencen con la palabra `List`, incluya la siguiente acción:

```
"Action": "osis:List*"
```

Para ver ejemplos de políticas basadas en identidad de OpenSearch Ingestion, consulte Ejemplos de políticas basadas en identidades para OpenSearch sin servidor.

Recursos de políticas de OpenSearch Ingestion

Compatibilidad con los recursos de políticas: sí

Los administradores pueden utilizar las políticas JSON de AWS para especificar quién tiene acceso a qué. Es decir, qué entidad principal puede realizar acciones en qué recursos y en qué condiciones.

El elemento `Resource` de la política JSON especifica el objeto u objetos a los que se aplica la acción. Como práctica recomendada, especifique un recurso utilizando el Nombre de recurso de Amazon (ARN). Para las acciones que no admiten permisos por recurso, utilice un carácter comodín (*) para indicar que la instrucción se aplica a todos los recursos.

```
"Resource": "*"
```

Claves de condición de política de Amazon OpenSearch Ingestion

Compatibilidad con claves de condición de políticas específicas del servicio: no

Los administradores pueden utilizar las políticas JSON de AWS para especificar quién tiene acceso a qué. Es decir, qué entidad principal puede realizar acciones en qué recursos y en qué condiciones.

El elemento `Condition` especifica cuándo se ejecutan las instrucciones en función de criterios definidos. Puede crear expresiones condicionales que utilizan operadores de condición, tales como igual o menor que, para que la condición de la política coincida con los valores de la solicitud. Para ver todas las claves de condición globales de AWS, consulte Claves de contexto de condición globales de AWS en la Guía del usuario de IAM.

Para obtener una lista de las claves de condición de OpenSearch Ingestion, consulte Claves de condición para Amazon OpenSearch Ingestión en la Referencia de autorizaciones de servicio. Para obtener más información sobre las acciones y los recursos con los que puede utilizar una clave de condición, consulte Acciones definidas por Amazon OpenSearch Ingestion.

ABAC con OpenSearch Ingestion

Admite ABAC (etiquetas en las políticas): sí

El control de acceso basado en atributos (ABAC) es una estrategia de autorización que define permisos en función de atributos denominados etiquetas. Puede asociar etiquetas a entidades de IAM y recursos de AWS y, a continuación, diseñar políticas de ABAC para permitir operaciones cuando la etiqueta de la entidad principal coincida con la etiqueta del recurso.

Para controlar el acceso en función de etiquetas, debe proporcionar información de las etiquetas en el elemento de condición de una política utilizando las claves de condición `aws:ResourceTag/`*`key-name`*, `aws:RequestTag/`*`key-name`* o `aws:TagKeys`.

Si un servicio admite las tres claves de condición para cada tipo de recurso, el valor es Sí para el servicio. Si un servicio admite las tres claves de condición solo para algunos tipos de recursos, el valor es Parcial.

Para obtener más información sobre ABAC, consulte Definición de permisos con la autorización de ABAC en la Guía del usuario de IAM. Para ver un tutorial con los pasos para configurar ABAC, consulte Uso del control de acceso basado en atributos (ABAC) en la Guía del usuario de IAM.

Para obtener más información sobre el etiquetado de recursos OpenSearch Ingestion, consulte the section called “Etiquetado de canalizaciones”.

Uso de credenciales temporales con OpenSearch Ingestion

Compatibilidad con credenciales temporales: sí

Las credenciales temporales proporcionan acceso a corto plazo a los recursos de AWS y se crean automáticamente cuando se utiliza la federación o se cambia de rol. AWS recomienda que genere de forma dinámica credenciales temporales en lugar de usar claves de acceso a largo plazo. Para obtener más información, consulte Credenciales de seguridad temporales en IAM y Servicios de AWS que funcionan con IAM en la Guía del usuario de IAM.

Roles vinculados a servicios para OpenSearch Ingestion

Admite roles vinculados a servicios: sí

Un rol vinculado a servicios es un tipo de rol de servicio que está vinculado a un Servicio de AWS. El servicio puedes asumir el rol para realizar una acción en su nombre. Los roles vinculados a servicios aparecen en la Cuenta de AWS y son propiedad del servicio. Un administrador de IAM puede ver, pero no editar, los permisos de los roles vinculados a servicios.

OpenSearch Ingestion utiliza un rol vinculado a servicios denominado `AWSServiceRoleForAmazonOpenSearchIngestionService`. El rol vinculado al servicio denominado `AWSServiceRoleForOpensearchIngestionSelfManagedVpce` también está disponible para canalizaciones con puntos de conexión de VPC autogestionados. Para obtener más información sobre cómo crear o administrar roles vinculados a servicios de OpenSearch Ingestion, consulte the section called “Rol de creación de canalizaciones”.

Ejemplos de políticas basadas en identidades para OpenSearch Ingestion

De forma predeterminada, los usuarios y roles no tienen permiso para crear ni modificar recursos de OpenSearch Ingestion. Un administrador de IAM puede crear políticas de IAM para conceder permisos a los usuarios para realizar acciones en los recursos que necesitan.

Para obtener información acerca de cómo crear una política basada en identidades de IAM mediante el uso de estos documentos de políticas JSON de ejemplo, consulte Creación de políticas de IAM (consola) en la Guía del usuario de IAM.

Para obtener más información sobre las acciones y los tipos de recursos definidos por Amazon OpenSearch Ingestion, incluido el formato de los ARN para cada uno de los tipos de recursos, consulte Acciones, recursos y claves de condición para Amazon OpenSearch Ingestionen la Referencia de autorizaciones de servicio.

Temas

- Prácticas recomendadas sobre las políticas
- Uso de OpenSearch Ingestion en la consola
- Administración de canalizaciones de OpenSearch Ingestion
- Incorporación de datos en una canalización de OpenSearch Ingestion

Prácticas recomendadas sobre las políticas

Las políticas basadas en identidad son muy eficaces. Determinan si alguien puede crear, acceder o eliminar los recursos de OpenSearch Ingestion de su cuenta. Estas acciones pueden generar costos adicionales para su Cuenta de AWS. Siga estas directrices y recomendaciones al crear o editar políticas basadas en identidades:

Las políticas basadas en identidades determinan si alguien puede crear, acceder o eliminar recursos de OpenSearch Ingestion de su cuenta. Estas acciones pueden generar costos adicionales para su Cuenta de AWS. Siga estas directrices y recomendaciones al crear o editar políticas basadas en identidades:

- Comienza con las políticas administradas por AWSy continúa con los permisos de privilegio mínimo: a fin de comenzar a conceder permisos a los usuarios y las cargas de tarea, utiliza las políticas administradas por AWS, que conceden permisos para muchos casos de uso comunes. Están disponibles en su Cuenta de AWS. Se recomienda definir políticas gestionadas por el

cliente de AWS específicas para sus casos de uso a fin de reducir aún más los permisos. Con el fin de obtener más información, consulte las políticas administradas por AWS o las políticas administradas por AWS para funciones de tarea en la Guía de usuario de IAM.

- Aplique permisos de privilegio mínimo: cuando establezca permisos con políticas de IAM, conceda solo los permisos necesarios para realizar una tarea. Para ello, debe definir las acciones que se pueden llevar a cabo en determinados recursos en condiciones específicas, también conocidos como permisos de privilegios mínimos. Con el fin de obtener más información sobre el uso de IAM para aplicar permisos, consulte Políticas y permisos en IAM en la Guía del usuario de IAM.
- Utilice condiciones en las políticas de IAM para restringir aún más el acceso: puede agregar una condición a sus políticas para limitar el acceso a las acciones y los recursos. Por ejemplo, puedes escribir una condición de políticas para especificar que todas las solicitudes deben enviarse utilizando SSL. También puedes usar condiciones para conceder acceso a acciones de servicios si se emplean a través de un Servicio de AWS determinado como, por ejemplo, CloudFormation. Para obtener más información, consulte Elementos de la política de JSON de IAM: Condición en la Guía del usuario de IAM.
- Utiliza el analizador de acceso de IAM para validar las políticas de IAM con el fin de garantizar la seguridad y funcionalidad de los permisos: el analizador de acceso de IAM valida políticas nuevas y existentes para que respeten el lenguaje (JSON) de las políticas de IAM y las prácticas recomendadas de IAM. El analizador de acceso de IAM proporciona más de 100 verificaciones de políticas y recomendaciones procesables para ayudar a crear políticas seguras y funcionales. Para más información, consulte Validación de políticas con el Analizador de acceso de IAM en la Guía del usuario de IAM.
- Solicite la autenticación multifactor (MFA): si se encuentra en una situación en la que necesite usuarios raíz o de IAM en su Cuenta de AWS, active la MFA para obtener una mayor seguridad. Para exigir la MFA cuando se invoquen las operaciones de la API, añada condiciones de MFA a sus políticas. Para más información, consulte Acceso seguro a la API con MFA en la Guía del usuario de IAM.

Para obtener más información sobre las prácticas recomendadas de IAM, consulte Prácticas recomendadas de seguridad en IAM en la Guía del usuario de IAM.

Uso de OpenSearch Ingestion en la consola

Para acceder a OpenSearch Ingestion en la consola de OpenSearch Service, debe tener un conjunto mínimo de permisos. Estos permisos deben permitirle registrar y consultar los detalles acerca de los recursos de OpenSearch Ingestion en su cuenta de AWS. Si se crea una política basada en

identidades que es más restrictiva que los permisos necesarios mínimos, la consola no funcionará del modo esperado para las entidades (tales como roles de IAM) que tengan esa política.

No es necesario que conceda permisos mínimos para la consola a los usuarios que solo realizan llamadas a la AWS CLI o a la API de AWS. En su lugar, permite acceso únicamente a las acciones que coincidan con la operación de API que intenta realizar.

La siguiente política permite al usuario acceder a OpenSearch Ingestion desde la consola de OpenSearch Service:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Resource": "*",
            "Effect": "Allow",
            "Action": [
                "osis:ListPipelines",
                "osis:GetPipeline",
                "osis:ListPipelineBlueprints",
                "osis:GetPipelineBlueprint",
                "osis:GetPipelineChangeProgress"
            ]
        }
    ]
}
```

Como alternativa, puede utilizar la política the section called “Acceso de solo lectura a Amazon OpenSearchingestion” AWS gestionada, que concede acceso de solo lectura a todos los recursos de OpenSearch Ingestion por un Cuenta de AWS.

Administración de canalizaciones de OpenSearch Ingestion

Esta política es un ejemplo de la política de “administración de canalizaciones” que permite al usuario gestionar y administrar las canalizaciones de Amazon OpenSearch Ingestion. El usuario puede crear, ver y eliminar canalizaciones.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Resource": "arn:aws:osis:us-east-1:111122223333:pipeline/*",
            "Action": [
                "osis:CreatePipeline",
                "osis:DeletePipeline",
                "osis:UpdatePipeline",
                "osis:ValidatePipeline",
                "osis:StartPipeline",
                "osis:StopPipeline"
            ],
            "Effect": "Allow"
        },
        {
            "Resource": "*",
            "Action": [
                "osis:ListPipelines",
                "osis:GetPipeline",
                "osis:ListPipelineBlueprints",
                "osis:GetPipelineBlueprint",
                "osis:GetPipelineChangeProgress"
            ],
            "Effect": "Allow"
        }
    ]
}
```

Incorporación de datos en una canalización de OpenSearch Ingestion

Este ejemplo de política permite a un usuario u otra entidad incorporar datos a una canalización de Amazon OpenSearch Ingestion de su cuenta. El usuario no puede modificar las canalizaciones.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
```

```
        {
            "Resource": "arn:aws:osis:us-east-1:123456789012:pipeline/*",
            "Action": [
                "osis:Ingest"
            ],
            "Effect": "Allow"
        }
    ]
}
```

Registro de llamadas a la API de Amazon OpenSearch Ingestion mediante AWS CloudTrail

Amazon OpenSearch Ingestion se integra con AWS CloudTrail, un servicio que proporciona un registro de las acciones realizadas por un usuario, un rol o un servicio de AWS en OpenSearch Ingestion.

CloudTrail captura todas las llamadas de la API para OpenSearch Ingestion como eventos. Las llamadas capturadas incluyen llamadas desde la sección de OpenSearch Ingestion de la consola de OpenSearch Service y llamadas de código a las operaciones de la API de OpenSearch Ingestion.

Si crea un registro de seguimiento, puede habilitar la entrega continua de eventos de CloudTrail a un bucket de Amazon S3, incluidos los eventos para OpenSearch Ingestion. Si no configura un registro de seguimiento, puede ver los eventos más recientes en la consola de CloudTrail en el Historial de eventos.

Mediante la información recopilada por CloudTrail, puede determinar la solicitud que se realizó en OpenSearch Ingestion, la dirección IP desde la cual se realizó, quién la realizó y cuándo, etc.

Para obtener más información sobre CloudTrail, consulte la Guía del usuario de AWS CloudTrail.

Información de OpenSearch Ingestion en CloudTrail

CloudTrail se habilita en su Cuenta de AWS cuando se crea la cuenta. Cuando se produce una actividad en OpenSearch Ingestion, esa actividad se registra en un evento de CloudTrail junto con otros eventos de servicio de AWS en Historial de eventos. Puede ver, buscar y descargar eventos recientes en su Cuenta de AWS. Para más información, consulte Viewing events with CloudTrail Event history.

Para mantener un registro continuo de los eventos en su Cuenta de AWS, incluidos los eventos de OpenSearch Ingestion, cree un registro de seguimiento. Un registro de seguimiento permite a CloudTrail enviar archivos de registro a un bucket de Amazon S3. De forma predeterminada, cuando se crea un registro de seguimiento en la consola, el registro de seguimiento se aplica a todas las Regiones de AWS.

El registro de seguimiento registra los eventos de todas las regiones de la partición de AWS y envía los archivos de registro al bucket de Amazon S3 especificado. También es posible configurar otros servicios de AWS para analizar en profundidad y actuar en función de los datos de eventos recopilados en los registros de CloudTrail. Para más información, consulte los siguientes temas:

- Introducción a la creación de registros de seguimiento
- Servicios e integraciones compatibles con CloudTrail
- Configuración de notificaciones de Amazon SNS para CloudTrail
- Recibir archivos de registro de CloudTrail de varias regiones y Recibir archivos de registro de CloudTrail de varias cuentas

CloudTrail registra todas las acciones de OpenSearch Ingestion que se documentan en la referencia de API de OpenSearch Ingestion. Por ejemplo, las llamadas a las acciones `CreateCollection`, `ListCollections` y `DeleteCollection` generan entradas en los archivos de registro de CloudTrail.

Cada entrada de registro o evento contiene información sobre quién generó la solicitud. La información de identidad lo ayuda a determinar:

- Si la solicitud se realizó con credenciales de usuario de AWS Identity and Access Management (IAM) o credenciales de usuario raíz.
- Si la solicitud se realizó con credenciales de seguridad temporales de un rol o fue un usuario federado.
- Si la solicitud la realizó otro servicio de AWS.

Para más información, consulte Elemento userIdentity de CloudTrail.

Descripción de las entradas del archivo de registro de OpenSearch Ingestion

Un registro de seguimiento es una configuración que permite la entrega de eventos como archivos de registros en un bucket de Amazon S3 que especifique. Los archivos de registro de CloudTrail contienen una o varias entradas de registro.

Un evento representa una única solicitud desde cualquier origen. Incluye información sobre la acción solicitada, la fecha y la hora de la acción, los parámetros de la solicitud, etc. Los archivos de registro de CloudTrail no rastrean el orden en la pila de las llamadas públicas a la API, por lo que estas no aparecen en ningún orden específico.

En el siguiente ejemplo, se muestra una entrada de registro de CloudTrail que ilustra la acción `DeletePipeline`.


```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn":"arn:aws:iam::123456789012:user/test-user",
        "accountId": "123456789012",
        "accessKeyId": "access-key",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                "arn": "arn:aws:iam::123456789012:role/Admin",
                "accountId": "123456789012",
                "userName": "Admin"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2023-04-21T16:48:33Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2023-04-21T16:49:22Z",
    "eventSource": "osis.amazonaws.com",
    "eventName": "UpdatePipeline",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "123.456.789.012",
```

```
    "userAgent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36
 (KHTML, like Gecko) Chrome/112.0.0.0 Safari/537.36",
    "requestParameters": {
        "pipelineName": "my-pipeline",
        "pipelineConfigurationBody": "version: \"2\"\nlog-pipeline:\n  source:\n
 http:\n        path: \"/test/logs\"\n  processor:\n    - grok:\n        match:\n
        log: [ '%{COMMONAPACHELOG}' ]\n    - date:\n        from_time_received: true
\n        destination: \"@timestamp\"\n  sink:\n    - opensearch:\n        hosts:
 [ \"https://search-b5zd22mwxhggheqpj5ftslgyle.us-west-2.es.amazonaws.com\" ]\n
  index: \"apache_logs2\"\n        aws_sts_role_arn: \"arn:aws:iam::709387180454:role/
canary-bootstrap-OsisRole-J1BARLD26QKN\"\n        aws_region: \"us-west-2\"\n
 aws_sigv4: true\n"
    },
    "responseElements": {
        "pipeline": {
            "pipelineName": "my-pipeline",sourceIPAddress
            "pipelineArn": "arn:aws:osis:us-west-2:123456789012:pipeline/my-pipeline",
            "minUnits": 1,
            "maxUnits": 1,
            "status": "UPDATING",
            "statusReason": {
                "description": "An update was triggered for the pipeline. It is still
 available to ingest data."
            },
            "pipelineConfigurationBody": "version: \"2\"\nlog-pipeline:\n  source:\n
    http:\n        path: \"/test/logs\"\n  processor:\n    - grok:\n        match:
\n          log: [ '%{COMMONAPACHELOG}' ]\n    - date:\n        from_time_received:
 true\n        destination: \"@timestamp\"\n  sink:\n    - opensearch:\n        hosts:
 [ \"https://search-b5zd22mwxhggheqpj5ftslgyle.us-west-2.es.amazonaws.com\" ]\n
  index: \"apache_logs2\"\n        aws_sts_role_arn: \"arn:aws:iam::709387180454:role/
canary-bootstrap-OsisRole-J1BARLD26QKN\"\n        aws_region: \"us-west-2\"\n
 aws_sigv4: true\n",
            "createdAt": "Mar 29, 2023 1:03:44 PM",
            "lastUpdatedAt": "Apr 21, 2023 9:49:21 AM",
            "ingestEndpointUrls": [
                "my-pipeline-tu33ldsgdltgv7x7tjqiudvf7m.us-west-2.osis.amazonaws.com"
            ]
        }
    },
    "requestID": "12345678-1234-1234-1234-987654321098",
    "eventID": "12345678-1234-1234-1234-987654321098",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
```

```
    "recipientAccountId": "709387180454",
    "eventCategory": "Management",
    "tlsDetails": {
        "tlsVersion": "TLSv1.2",
        "cipherSuite": "ECDHE-RSA-AES128-GCM-SHA256",
        "clientProvidedHostHeader": "osis.us-west-2.amazonaws.com"
    },
    "sessionCredentialFromConsole": "true"
}
```

Etiquetado de canalizaciones de Amazon OpenSearch Ingestion

Las etiquetas permiten asignar información arbitraria a una canalización de Amazon OpenSearch Ingestion para poder categorizar y filtrar esa información. Una etiqueta es un elemento de metadatos que usted o AWS asigna a un recurso de AWS. Cada etiqueta consta de una clave y un valor. En el caso de etiquetas que usted asigna, debe definir la clave y el valor. Por ejemplo, puede definir la clave como `stage` y el valor de un recurso como `test`.

Las etiquetas lo ayudan a hacer lo siguiente:

- Identificar y organizar sus recursos de AWS. Muchos servicios de AWS admiten el etiquetado, por lo que puede asignar la misma etiqueta a los recursos de diferentes servicios para indicar que los recursos están relacionados. Por ejemplo, podría asignarle la misma etiqueta a una canalización de OpenSearch Ingestion que a un dominio de Amazon OpenSearch Service.
- Realizar un seguimiento de los costos de AWS. Estas etiquetas se activan en el panel de Administración de facturación y costos de AWS. AWS usa las etiquetas para clasificar los costos y enviar un informe mensual de asignación de costos. Para obtener más información, consulte Uso de etiquetas de asignación de costos en la Guía del usuario de AWS Billing.
- Restrinja el acceso a las canalizaciones mediante el control de acceso basado en atributos. Para obtener más información, consulte Controlling access based on tag keys en la Guía del usuario de IAM.

En OpenSearch Ingestion, el recurso principal es una canalización. Puede utilizar la consola de OpenSearch Service, la CLI AWS, las API de OpenSearch Ingestion o los SDK de AWS para agregar, administrar y eliminar etiquetas de una canalización.

Temas

- Permisos necesarios

- Uso de etiquetas (consola)
- Uso de etiquetas (AWS CLI)

Permisos necesarios

OpenSearch Ingestion utiliza los siguientes permisos AWS Identity and Access Management Access Analyzer (IAM) para etiquetar las canalizaciones:

- `osis:TagResource`
- `osis:ListTagsForResource`
- `osis:UntagResource`

Para obtener más información sobre cada permiso, consulte Acciones, recursos y claves de condición de OpenSearch Ingestion en la Referencia de autorizaciones de servicio.

Uso de etiquetas (consola)

La consola es la forma más sencilla para etiquetar una canalización.

Cómo crear una etiqueta

1. Inicie sesión en la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. Seleccione Canalizaciones en el panel de navegación izquierdo.
3. Seleccione una canalización a la que desee agregar etiquetas y vaya a la pestaña Etiquetas.
4. Seleccione Administrar y Agregar nueva etiqueta.
5. Introduzca una clave de etiqueta y un valor opcional.
6. Seleccione guardar.

Para eliminar una etiqueta, siga los mismos pasos y elija Quitar en la página Administrar etiquetas.

Para obtener más información sobre cómo utilizar la consola para trabajar con etiquetas, consulte Tag Editor en la Guía de introducción a la consola de administración de AWS.

Uso de etiquetas (AWS CLI)

Para etiquetar una canalización mediante el AWS CLI, envíe una solicitud `TagResource`:

```
aws osis tag-resource
  --arn arn:aws:osis:us-east-1:123456789012:pipeline/my-pipeline
  --tags Key=service,Value=osis Key=source,Value=otel
```

Elimine las etiquetas de una canalización mediante el comando `UntagResource`:

```
aws osis untag-resource
  --arn arn:aws:osis:us-east-1:123456789012:pipeline/my-pipeline
  --tag-keys service
```

Consulte las etiquetas existentes para una canalización con el comando `ListTagsForResource`:

```
aws osis list-tags-for-resource
  --arn arn:aws:osis:us-east-1:123456789012:pipeline/my-pipeline
```

Registro y monitoreo de Amazon OpenSearch Ingestion con Amazon CloudWatch

Amazon OpenSearch Ingestion publica métricas y registros en Amazon CloudWatch.

Temas

- Monitoreo de registros de canalización
- Monitoreo de métricas de canalización

Monitoreo de registros de canalización

Puede habilitar el registro de las canalizaciones de Amazon OpenSearch Ingestion para mostrar los mensajes de error y advertencia que se generen durante las operaciones de canalización y la actividad de incorporación. Amazon OpenSearch Ingestion publica todos los registros en Registros de Amazon CloudWatch. Los registros de CloudWatch pueden supervisar la información de los archivos de registro y enviarle una notificación cuando se alcancen determinados umbrales. También se pueden archivar los datos del registro en un almacenamiento de larga duración. Para obtener más información, consulte la Guía del usuario de Registros de Amazon CloudWatch.

Los registros de OpenSearch Ingestion pueden indicar un error en el procesamiento de solicitudes, errores de autenticación desde el origen hasta el receptor y otras advertencias que pueden resultar

útiles para solucionar problemas. Para sus registros, OpenSearch Ingestion utiliza los niveles de registro de `INFO`, `WARN`, `ERROR` y `FATAL`. Recomendamos habilitar la publicación de registros en todas las canalizaciones.

Permisos necesarios

Para habilitar OpenSearch Ingestion a fin de enviar registros a Registros de CloudWatch, debe iniciar sesión como usuario con ciertos permisos de IAM.

Necesita los siguientes permisos de Registros de CloudWatch para crear y actualizar los recursos de entrega de registros:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Resource": "*",
            "Action": [
                "logs:CreateLogDelivery",
                "logs:PutResourcePolicy",
                "logs:UpdateLogDelivery",
                "logs:DeleteLogDelivery",
                "logs:DescribeResourcePolicies",
                "logs:GetLogDelivery",
                "logs:ListLogDeliveries"
            ]
        }
    ]
}
```

Habilitación de la publicación de registros

Puede habilitar la publicación de registros en las canalizaciones existentes o mientras crea una canalización. Para ver los pasos para habilitar la publicación de registros durante la creación de la canalización, consulte the section called “Creación de canalizaciones”.

Consola

Cómo habilitar la publicación de registros en una canalización existente

1. Inicie sesión en la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. Seleccione Canalizaciones en el panel de navegación izquierdo.
3. Abra la canalización en la que desea habilitar los registros y, a continuación, seleccione Acciones y Editar las opciones de publicación de registros.
4. Habilite Publicar en Registros de CloudWatch.
5. Cree un grupo de registro nuevo o seleccione uno que ya exista. Se recomienda formatear el nombre como una ruta, por ejemplo`/aws/`**`vendedlogs`**`/OpenSearchIngestion/`*`pipeline-name`*`/audit-logs`. Este formato facilita la aplicación de una política de acceso a CloudWatch que concede permisos a todos los grupos de registro en una ruta específica, como `/aws/vendedlogs/OpenSearchIngestion`.

   > **Important**
   >
   > Debe incluir el prefijo `vendedlogs` en el nombre del grupo de registro; de lo contrario, se producirá un error en la creación.

6. Seleccione Save.

CLI

Para habilitar la publicación de registros mediante el AWS CLI, envíe la siguiente solicitud:

```
aws osis update-pipeline \
  --pipeline-name my-pipeline \
  --log-publishing-options  IsLoggingEnabled=true,CloudWatchLogDestination={LogGroup="/
aws/vendedlogs/OpenSearchIngestion/pipeline-name"}
```

Monitoreo de métricas de canalización

Puede monitorear las canalizaciones de Amazon OpenSearch Ingestion mediante Amazon CloudWatch, que recopila y procesa los datos sin procesar y los convierte en métricas legibles y casi en tiempo real. Estas estadísticas se mantienen durante 15 meses, de forma que pueda obtener acceso a información histórica y disponer de una mejor perspectiva sobre el rendimiento de su

aplicación web o servicio. También puede establecer alarmas que vigilen determinados umbrales y enviar notificaciones o realizar acciones cuando se cumplan dichos umbrales. Para obtener más información, consulte la Guía del usuario de Amazon CloudWatch.

La consola de OpenSearch Ingestion muestra una serie de gráficos basados en los datos sin procesar de CloudWatch en la pestaña Rendimiento de cada canalización.

OpenSearch Ingestion informa de las métricas de la mayoría de los complementos compatibles. Si algunos complementos no tienen su propia tabla a continuación, significa que no muestran ninguna métrica específica de los complementos. Las métricas de canalizaciones se publican en el espacio de nombres de `AWS/OSIS`.

Temas

- Métricas comunes
- Métricas de búfer
- Métricas de la V4 de Signature
- Métricas del búfer de bloqueo limitado
- Métricas de origen de trazas de Otel
- Métricas de OTel: métricas de origen
- Métricas de Http
- Métricas de S3
- Métricas agrupadas
- Métricas de fecha
- métricas de Lambda
- Métricas de Grok
- Métricas sin formato de trazas de OTel
- Métricas del grupo de trazas de OTel
- Métricas de estado del mapa de servicio
- Métricas de OpenSearch
- Métricas del sistema y de medición

Métricas comunes

Las siguientes métricas son comunes a todos los procesadores y receptores.

Cada métrica lleva el nombre de la subcanalización y el nombre del complemento, con el formato < *sub_pipeline_name* >< *plugin*><*metric_name* >. Por ejemplo, el nombre completo de la métrica `recordsIn.count` de una subcanalización denominada `my-pipeline` y la fecha del procesador sería `my-pipeline.date.recordsIn.count`.

Sufijo de la métrica	Descripción
`recordsIn.count`	El ingreso de registros a un componente de canalización. Esta métrica se aplica a procesadores y receptores. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`recordsOut.count`	El egreso de registros de un componente de canalización. Esta métrica se aplica a procesadores y orígenes. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`timeElapsed.count`	Un recuento de los puntos de datos registrados durante la ejecución de un componente de canalización. Esta métrica se aplica a procesadores y receptores. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`timeElapsed.sum`	El tiempo total transcurrido durante la ejecución de un componente de canalización. Esta métrica se aplica a procesadores y receptores, en milisegundos. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`timeElapsed.max`	El tiempo máximo transcurrido durante la ejecución de un componente de canalización. Esta métrica se aplica a procesadores y receptores, en milisegundos.

Sufijo de la métrica	Descripción
	Estadísticas pertinentes: máximo Dimensión: `PipelineName`

Métricas de búfer

Las siguientes métricas se aplican al búfer de bloqueo limitado predeterminado que OpenSearch Ingestion configura automáticamente para todas las canalizaciones.

Cada métrica está precedida por el nombre de la subcanalización y el nombre del búfer, con el formato <*sub_pipeline_name*><*buffer_name*><*metric_name*>. Por ejemplo, el nombre completo de la métrica `recordsWritten.count` de una subcanalización denominada `my-pipeline` sería `my-pipeline.BlockingBuffer.recordsWritten.count`.

Sufijo de la métrica	Descripción
`recordsWritten.count`	El número de registros escritos en un búfer. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`recordsRead.count`	El número de registros leídos de un búfer. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`recordsInFlight.value`	El número de registros sin comprobar de un búfer. Estadísticas pertinentes: promedio Dimensión: `PipelineName`
`recordsInBuffer.value`	El número de registros que hay actualmente en un búfer. Estadísticas pertinentes: promedio Dimensión: `PipelineName`

Sufijo de la métrica	Descripción
`recordsProcessed.count`	El número de registros leídos de un búfer y procesados por una canalización. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`recordsWriteFailed.count`	El número de registros que la canalización no ha podido escribir en el receptor. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`writeTimeElapsed.count`	Recuento de puntos de datos registrados mientras se escribe en un búfer. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`writeTimeElapsed.sum`	El tiempo total transcurrido durante la escritura en un búfer, en milisegundos. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`writeTimeElapsed.max`	El tiempo máximo transcurrido durante la escritura en un búfer, en milisegundos. Estadísticas pertinentes: máximo Dimensión: `PipelineName`
`writeTimeouts.count`	El recuento de tiempos de espera de escritura en un búfer. Estadísticas pertinentes: suma Dimensión: `PipelineName`

Sufijo de la métrica	Descripción
`readTimeElapsed.count`	Recuento de puntos de datos registrados mientras se lee de un búfer. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`readTimeElapsed.sum`	El tiempo total transcurrido mientras se lee de un búfer, en milisegundos. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`readTimeElapsed.max`	El tiempo máximo transcurrido mientras se lee de un búfer, en milisegundos. Estadísticas pertinentes: máximo Dimensión: `PipelineName`
`checkpointTimeElapsed.count`	Un recuento de los puntos de datos registrados durante el punto de control. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`checkpointTimeElapsed.sum`	El tiempo total transcurrido durante el punto de control, en milisegundos. Estadísticas pertinentes: suma Dimensión: `PipelineName`

Sufijo de la métrica	Descripción
`checkpointTimeElapsed.max`	El tiempo máximo transcurrido durante el punto de control, en milisegundos. Estadísticas pertinentes: máximo Dimensión: `PipelineName`

Métricas de la V4 de Signature

Las siguientes métricas se aplican al punto de conexión de ingesta de una canalización y están asociadas a los complementos de origen (`http`, `otel_trace`, y `otel_metrics`). Todas las solicitudes al punto de conexión de ingesta deben firmarse con la versión 4 de Signature. Estas métricas pueden ayudarle a identificar los problemas de autorización al conectarse a su canalización o a confirmar que se está autenticando correctamente.

Cada métrica lleva el nombre de la subcanalización como prefijo y `osis_sigv4_auth`. Por ejemplo, *`sub_pipeline_name`*`.osis_sigv4_auth.httpAuthSuccess.count`.

Sufijo de la métrica	Descripción
`httpAuthSuccess.count`	El número de solicitudes de Signature V4 que se han enviado correctamente a la canalización. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`httpAuthFailure.count`	El número de solicitudes fallidas de Signature V4 enviadas a la canalización. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`httpAuthServerError.count`	El número de solicitudes de Signature V4 enviadas a la canalización que arrojaron errores en el servidor. Estadísticas pertinentes: suma

Sufijo de la métrica	Descripción
	Dimensión: `PipelineName`

Métricas del búfer de bloqueo limitado

Las siguientes métricas se aplican al búfer de bloqueo limitado. Cada métrica lleva el nombre de la subcanalización como prefijo y `BlockingBuffer`. Por ejemplo, *`sub_pipeline_name`*`.BlockingBuffer.bufferUsage.value`.

Sufijo de la métrica	Descripción
`bufferUsage.value`	El porcentaje de uso del `buffer_size` se basa en el número de registros del búfer. `buffer_size` represent a el número máximo de registros escritos en el búfer, así como los registros en movimiento que no se han comprobado. Estadísticas pertinentes: promedio Dimensión: `PipelineName`

Métricas de origen de trazas de Otel

Las siguientes métricas se aplican al orígen de trazas de OTel. Cada métrica lleva el nombre de la subcanalización como prefijo y `otel_trace_source`. Por ejemplo, *`sub_pipeline_name`*`.otel_trace_source.requestTimeouts.count`.

Sufijo de la métrica	Descripción
`requestTimeouts.count`	El número de solicitudes que agotaron el tiempo de espera. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`requestsReceived.count`	El número de solicitudes recibidas por el complemento.

Sufijo de la métrica	Descripción
	Estadísticas pertinentes: suma Dimensión: `PipelineName`
`successRequests.count`	El número de solicitudes que el complemento procesó correctamente. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`badRequests.count`	El número de solicitudes con un formato inválido procesadas por el complemento. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`requestsTooLarge.count`	El número de solicitudes cuya cantidad de intervalos en el contenido es superior a la capacidad del búfer. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`internalServerError.count`	El número de solicitudes procesadas por el complemento con un tipo de excepción personalizado. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`requestProcessDuration.count`	Un recuento de los puntos de datos registrados durante el procesamiento de las solicitudes por parte del complemento. Estadísticas pertinentes: suma Dimensión: `PipelineName`

Sufijo de la métrica	Descripción
`requestProcessDuration.sum`	La latencia total de las solicitudes procesadas por el complemento, en milisegundos. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`requestProcessDuration.max`	La latencia máxima de las solicitudes procesadas por el complemento, en milisegundos. Estadísticas pertinentes: máximo Dimensión: `PipelineName`
`payloadSize.count`	Un recuento de la distribución de los tamaños de carga útil de las solicitudes entrantes, en bytes. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`payloadSize.sum`	La distribución total de los tamaños de carga útil de las solicitudes entrantes, en bytes. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`payloadSize.max`	La distribución máxima de los tamaños de carga útil de las solicitudes entrantes, en bytes. Estadísticas pertinentes: máximo Dimensión: `PipelineName`

Métricas de OTel: métricas de origen

Las siguientes métricas se aplican al origen de las métricas de OTel. Cada métrica lleva el nombre de la subcanalización como prefijo y `otel_metrics_source`. Por ejemplo, *`sub_pipeline_name`*`.otel_metrics_source.requestTimeouts.count`.

Sufijo de la métrica	Descripción
`requestTimeouts.count`	El número total de solicitudes al complemento que agotaron el tiempo de espera. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`requestsReceived.count`	El número total de solicitudes recibidas por el complemento. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`successRequests.count`	El número de solicitudes procesadas correctamente (código de estado de respuesta de 200) por el complemento. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`requestProcessDuration.count`	Un recuento de la latencia de las solicitudes procesadas por el complemento, en segundos. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`requestProcessDuration.sum`	La latencia total de las solicitudes procesadas por el complemento, en milisegundos. Estadísticas pertinentes: suma

Sufijo de la métrica	Descripción
	Dimensión: `PipelineName`
`requestProcessDuration.max`	La latencia máxima de las solicitudes procesadas por el complemento, en milisegundos. Estadísticas pertinentes: máximo Dimensión: `PipelineName`
`payloadSize.count`	Un recuento de la distribución de los tamaños de carga útil de las solicitudes entrantes, en bytes. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`payloadSize.sum`	La distribución total de los tamaños de carga útil de las solicitudes entrantes, en bytes. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`payloadSize.max`	La distribución máxima de los tamaños de carga útil de las solicitudes entrantes, en bytes. Estadísticas pertinentes: máximo Dimensión: `PipelineName`

Métricas de Http

Las siguientes métricas se aplican al origen de HTTP. Cada métrica lleva el nombre de la subcanalización como prefijo y `http`. Por ejemplo, *`sub_pipeline_name`*`.http.requestsReceived.count`.

Sufijo de la métrica	Descripción
`requestsReceived.count`	El número de solicitudes recibidas por el punto de conexión de `/log/ingest` . Estadísticas pertinentes: suma Dimensión: `PipelineName`
`requestsRejected.count`	El número de solicitudes rechazadas (código de estado de respuesta de 429) por el complemento. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`successRequests.count`	El número de solicitudes procesadas correctamente (código de estado de respuesta de 200) por el complemento. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`badRequests.count`	El número de solicitudes con un tipo o formato de contenido no válido (código de estado de respuesta de 400) procesadas por el complemento. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`requestTimeouts.count`	El número de solicitudes que agotan el tiempo de espera en el servidor origen de HTTP (código de estado de respuesta 415). Estadísticas pertinentes: suma Dimensión: `PipelineName`

Sufijo de la métrica	Descripción
`requestsTooLarge.count`	El número de solicitudes cuyo tamaño de eventos en el contenido es superior a la capacidad del búfer (código de estado de respuesta de 413). Estadísticas pertinentes: suma Dimensión: `PipelineName`
`internalServerError.count`	El número de solicitudes procesadas por el complemento con un tipo de excepción personalizado (código de estado de respuesta de 500). Estadísticas pertinentes: suma Dimensión: `PipelineName`
`requestProcessDuration.count`	Un recuento de la latencia de las solicitudes procesadas por el complemento, en segundos. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`requestProcessDuration.sum`	La latencia total de las solicitudes procesadas por el complemento, en milisegundos. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`requestProcessDuration.max`	La latencia máxima de las solicitudes procesadas por el complemento, en milisegundos. Estadísticas pertinentes: máximo Dimensión: `PipelineName`

Sufijo de la métrica	Descripción
`payloadSize.count`	Un recuento de la distribución de los tamaños de carga útil de las solicitudes entrantes, en bytes. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`payloadSize.sum`	La distribución total de los tamaños de carga útil de las solicitudes entrantes, en bytes. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`payloadSize.max`	La distribución máxima de los tamaños de carga útil de las solicitudes entrantes, en bytes. Estadísticas pertinentes: máximo Dimensión: `PipelineName`

Métricas de S3

Las siguientes métricas se aplican a la fuente de S3. Cada métrica lleva el nombre de la subcanalización como prefijo y `s3`. Por ejemplo, *`sub_pipeline_name`*`.s3.s3ObjectsFailed.count`.

Sufijo de la métrica	Descripción
`s3ObjectsFailed.count`	El número total de objetos S3 que el complemento no ha podido leer. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`s3ObjectsNotFound.count`	El número de objetos de S3 que el complemento no ha podido leer debido a un error `Not Found` de S3. Estas

Sufijo de la métrica	Descripción
	métricas también se tienen en cuenta para la métrica `s3ObjectsFailed` . Estadísticas pertinentes: suma Dimensión: `PipelineName`
`s3ObjectsAccessDenied.count`	El número de objetos de S3 que el complemento no ha podido leer debido a un error `Access Denied` o `Forbidden` de S3. Estas métricas también se tienen en cuenta para la métrica `s3ObjectsFailed` . Estadísticas pertinentes: suma Dimensión: `PipelineName`
`s3ObjectReadTimeElapsed.count`	El tiempo que tarda el complemento en realizar una solicitud GET para un objeto de S3, analizarla y escribir los eventos en el búfer. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`s3ObjectReadTimeElapsed.sum`	El tiempo total que tarda el complemento en realizar una solicitud GET para un objeto de S3, analizarla y escribir los eventos en el búfer, en milisegundos. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`s3ObjectReadTimeElapsed.max`	El tiempo máximo que tarda el complemento en realizar una solicitud GET para un objeto de S3, analizarla y escribir los eventos en el búfer, en milisegundos. Estadísticas pertinentes: máximo Dimensión: `PipelineName`

Sufijo de la métrica	Descripción
`s3ObjectSizeBytes.count`	El recuento de la distribución de los tamaños de los objetos de S3, en bytes. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`s3ObjectSizeBytes.sum`	La distribución total de los tamaños de los objetos de S3, en bytes. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`s3ObjectSizeBytes.max`	La distribución máxima de los tamaños de los objetos de S3, en bytes. Estadísticas pertinentes: máximo Dimensión: `PipelineName`
`s3ObjectProcessedBytes.count`	El recuento de la distribución de los objetos de S3 procesados por el complemento, en bytes. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`s3ObjectProcessedBytes.sum`	La distribución total de los objetos de S3 procesados por el complemento, en bytes. Estadísticas pertinentes: suma Dimensión: `PipelineName`

Sufijo de la métrica	Descripción
`s3ObjectProcessedBytes.max`	La distribución máxima de los objetos de S3 procesados por el complemento, en bytes. Estadísticas pertinentes: máximo Dimensión: `PipelineName`
`s3ObjectsEvents.count`	El recuento de la distribución de los eventos de S3 recibidos por el complemento. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`s3ObjectsEvents.sum`	La distribución total de los eventos de S3 recibidos por el complemento. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`s3ObjectsEvents.max`	La distribución máxima de los eventos de S3 recibidos por el complemento. Estadísticas pertinentes: máximo Dimensión: `PipelineName`
`sqsMessageDelay.count`	Un recuento de puntos de datos registrados mientras S3 registra el tiempo de un evento desde la creación de un objeto hasta el momento en que se analiza por completo. Estadísticas pertinentes: suma Dimensión: `PipelineName`

Sufijo de la métrica	Descripción
`sqsMessageDelay.sum`	La cantidad de tiempo total entre el momento en que S3 registra un evento para la creación de un objeto y el momento en que se analiza por completo, en milisegundos. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`sqsMessageDelay.max`	La cantidad máxima de tiempo entre el momento en que S3 registra un evento para la creación de un objeto y el momento en que se analiza por completo, en milisegundos. Estadísticas pertinentes: máximo Dimensión: `PipelineName`
`s3ObjectsSucceeded.count`	El número de objetos de S3 que el complemento ha leído correctamente. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`sqsMessagesReceived.count`	El número de mensajes de Amazon SQS recibidos de la cola por el complemento. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`sqsMessagesDeleted.count`	El número de mensajes de Amazon SQS eliminados de la cola por el complemento. Estadísticas pertinentes: suma Dimensión: `PipelineName`

Sufijo de la métrica	Descripción
`sqsMessagesFailed.count`	El número de mensajes de Amazon SQS que el complemento no ha podido analizar. Estadísticas pertinentes: suma Dimensión: `PipelineName`

Métricas agrupadas

Las siguientes métricas se aplican al procesador Aggregate. Cada métrica lleva el nombre de la subcanalización como prefijo y `aggregate`. Por ejemplo, *`sub_pipeline_name`*`.aggregate.actionHandleEventsOut.count`.

Sufijo de la métrica	Descripción
`actionHandleEventsOut.count`	El número de eventos que se devolvieron de la llamada `handleEvent` a la acción configurada. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`actionHandleEventsDropped.count`	El número de eventos que se devolvieron de la llamada `handleEvent` a la acción configurada. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`actionHandleEventsProcessingErrors.count`	El número de llamadas realizadas a `handleEvent` para la acción configurada que provocaron un error. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`actionConcludeGroupEventsOut.count`	El número de eventos que se devolvieron de la llamada `concludeGroup` a la acción configurada.

Sufijo de la métrica	Descripción
	Estadísticas pertinentes: suma Dimensión: `PipelineName`
`actionConcludeGroupEventsDropped.count`	El número de eventos que no se devolvieron de la llamada `condludeGroup` a la acción configurada. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`actionConcludeGroupEventsProcessingErrors.count`	El número de llamadas realizadas a `concludeGroup` para la acción configurada que provocaron un error. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`currentAggregateGroups.value`	El número actual de grupos. Este indicador disminuye cuando se concluyen los grupos y aumenta cuando un evento inicia la creación de un grupo nuevo. Estadísticas pertinentes: promedio Dimensión: `PipelineName`

Métricas de fecha

Las siguientes métricas se aplican al procesador Date. Cada métrica lleva el nombre de la subcanalización como prefijo y `date`. Por ejemplo, *`sub_pipeline_name`*`.date.dateProcessingMatchSuccess.count`.

Sufijo de la métrica	Descripción
`dateProcessingMatchSuccess.count`	El número de registros que coinciden con al menos uno de los patrones especificados en la opción `match` de configuración.

Sufijo de la métrica	Descripción
	Estadísticas pertinentes: suma Dimensión: `PipelineName`
`dateProcessingMatchFailure.count`	El número de registros que no coinciden con ninguno de los patrones especificados en la opción `match` de configuración. Estadísticas pertinentes: suma Dimensión: `PipelineName`

métricas de Lambda

Las siguientes métricas se aplican al procesador de AWS Lambda. Cada métrica lleva el nombre de la subcanalización como prefijo y `lambda`. Por ejemplo, *`sub_pipeline_name`*`.lambda.recordsSuccessfullySentToLambda.count`.

Sufijo de la métrica	Descripción
`recordsSuccessfullySentToLambda.count`	Número de registros procesados correctamente por la función de Lambda. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`recordsFailedToSendToLambda.count`	Número de registros que no se pudieron enviar a la función de Lambda. Estadísticas pertinentes: suma Dimensión: `PipelineName`

`lambdaFunctionLatency.avg`
 	
`lambdaFunctionLatency.max`
	La latencia de las invocaciones de funciones de Lambda.
	
	Estadísticas pertinentes: promedio y máximo

Sufijo de la métrica	Descripción
	Dimensión: `PipelineName`
`numberOfRequestsSucceeded.count`	El número total de solicitudes de invocaciones de Lambda correctas. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`numberOfRequestsFailed.count`	El número total de errores de solicitud de invocación de Lambda. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`requestPayloadSize.avg`	El tamaño de las cargas de solicitud enviadas a Lambda. Estadísticas pertinentes: promedio Dimensión: `PipelineName`
`responsePayloadSize.avg`	El tamaño de las cargas útiles de respuesta recibidas de Lambda. Estadísticas pertinentes: promedio Dimensión: `PipelineName`

Métricas de Grok

Las siguientes métricas se aplican al procesador Grok. Cada métrica lleva el nombre de la subcanalización como prefijo y `grok`. Por ejemplo, *`sub_pipeline_name`*`.grok.grokProcessingMatch.count`.

Sufijo de la métrica	Descripción
`grokProcessingMatch.count`	El número de registros que encontraron al menos un patrón coincidente en la opción `match` de configuración. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`grokProcessingMismatch.count`	El número de registros que no coinciden con ninguno de los patrones especificados en la opción `match` de configuración. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`grokProcessingErrors.count`	El número de errores de procesamiento de registros. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`grokProcessingTimeouts.count`	El número de registros cuyo tiempo de espera se agotó durante la coincidencia. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`grokProcessingTime.count`	Un recuento de los puntos de datos registrados mientras un registro individual coincidía con los patrones de la opción `match` de configuración. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`grokProcessingTime.sum`	El tiempo total que tarda cada registro individual en coincidir con los patrones de la opción `match` de configuración, en milisegundos.

Sufijo de la métrica	Descripción
	Estadísticas pertinentes: suma Dimensión: `PipelineName`
`grokProcessingTime.max`	El tiempo máximo que tarda cada registro individual en coincidir con los patrones de la opción `match` de configuración, en milisegundos. Estadísticas pertinentes: máximo Dimensión: `PipelineName`

Métricas sin formato de trazas de OTel

Las siguientes métricas se aplican al procesador de trazas sin procesar de OTel. Cada métrica lleva el nombre de la subcanalización como prefijo y `otel_trace_raw`. Por ejemplo, *`sub_pipeline_name`*`.otel_trace_raw.traceGroupCacheCount.value`.

Sufijo de la métrica	Descripción
`traceGroupCacheCount.value`	El número de grupos de trazas en la memoria caché de grupos de trazas. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`spanSetCount.value`	El número de conjuntos de intervalos de la colección de conjuntos de intervalos. Estadísticas pertinentes: suma Dimensión: `PipelineName`

Métricas del grupo de trazas de OTel

Las siguientes métricas se aplican al procesador de grupo de trazas de OTel. Cada métrica lleva el nombre de la subcanalización como prefijo y `otel_trace_group`. Por ejemplo, *`sub_pipeline_name`*`.otel_trace_group.recordsInMissingTraceGroup.count`.

Sufijo de la métrica	Descripción
`recordsInMissingTraceGroup.count`	El número de registros de ingreso que faltan en los campos del grupo de trazas. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`recordsOutFixedTraceGroup.count`	El número de registros de egreso con campos de grupos de trazas que se completaron correctamente. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`recordsOutMissingTraceGroup.count`	El número de registros de egreso que faltan en los campos del grupo de trazas. Estadísticas pertinentes: suma Dimensión: `PipelineName`

Métricas de estado del mapa de servicio

Las siguientes métricas se aplican al procesador del estado del mapa de servicio. Cada métrica lleva el nombre de la subcanalización como prefijo y `service-map-stateful`. Por ejemplo, *`sub_pipeline_name`*`.service-map-stateful.spansDbSize.count`.

Sufijo de la métrica	Descripción
`spansDbSize.value`	Los tamaños de los bytes en memoria de los intervalos en MapDB durante la duración de la ventana actual y anterior.

Sufijo de la métrica	Descripción
	Estadísticas pertinentes: promedio Dimensión: `PipelineName`
`traceGroupDbSize.value`	Los tamaños de los bytes en memoria de los grupos de trazas en MapDB durante la duración de la ventana actual y anterior. Estadísticas pertinentes: promedio Dimensión: `PipelineName`
`spansDbCount.value`	El recuento de intervalos en MapDB durante la duración de la ventana actual y anterior. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`traceGroupDbCount.value`	El recuento de grupos de trazas en MapDB durante la duración de la ventana actual y anterior. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`relationshipCount.value`	El recuento de las relaciones almacenadas en la duración de la ventana actual y anterior. Estadísticas pertinentes: suma Dimensión: `PipelineName`

Métricas de OpenSearch

Las siguientes métricas se aplican al receptor de OpenSearch. Cada métrica lleva el nombre de la subcanalización como prefijo y `opensearch`. Por ejemplo, *`sub_pipeline_name`*`.opensearch.bulkRequestErrors.count`.

Sufijo de la métrica	Descripción
`bulkRequestErrors.count`	El número total de errores encontrados al enviar solicitud es masivas. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`documentsSuccess.count`	El número de documentos enviados correctamente a OpenSearch Service mediante solicitud masiva, incluidos los reintentos. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`documentsSuccessFirstAttempt.count`	El número de documentos enviados correctamente a OpenSearch Service mediante solicitud masiva en el primer intento. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`documentErrors.count`	El número de documentos que no se pudieron enviar mediante solicitudes masivas. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`bulkRequestFailed.count`	El número de solicitudes masivas que produjeron un error. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`bulkRequestNumberOfRetries.count`	El número de reintentos de solicitudes masivas fallidas. Estadísticas pertinentes: suma

Sufijo de la métrica	Descripción
	Dimensión: `PipelineName`
`bulkBadRequestErrors.count`	El número de errores `Bad Request` encontrados al enviar solicitudes masivas. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`bulkRequestNotAllowedErrors.count`	El número de errores `Request Not Allowed` encontrados al enviar solicitudes masivas. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`bulkRequestInvalidInputErrors.count`	El número de errores `Invalid Input` encontrados al enviar solicitudes masivas. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`bulkRequestNotFoundErrors.count`	El número de errores `Request Not Found` encontrados al enviar solicitudes masivas. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`bulkRequestTimeoutErrors.count`	El número de errores `Request Timeout` encontrados al enviar solicitudes masivas. Estadísticas pertinentes: suma Dimensión: `PipelineName`

Sufijo de la métrica	Descripción
`bulkRequestServerErrors.count`	El número de errores `Server Error` encontrados al enviar solicitudes masivas. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`bulkRequestSizeBytes.count`	Un recuento de la distribución de los tamaños de carga útil de las solicitudes masivas, en bytes. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`bulkRequestSizeBytes.sum`	La distribución total de los tamaños de carga útil de las solicitudes masivas, en bytes. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`bulkRequestSizeBytes.max`	La distribución máxima de los tamaños de carga útil de las solicitudes masivas, en bytes. Estadísticas pertinentes: máximo Dimensión: `PipelineName`
`bulkRequestLatency.count`	Un recuento de los puntos de datos registrados mientras se envían las solicitudes al complemento, incluidos los reintentos. Estadísticas pertinentes: suma Dimensión: `PipelineName`

Sufijo de la métrica	Descripción
`bulkRequestLatency.sum`	La latencia total de las solicitudes enviadas al complemento, incluidos los reintentos, en milisegundos. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`bulkRequestLatency.max`	La latencia máxima de las solicitudes enviadas al complemento, incluidos los reintentos, en milisegundos. Estadísticas pertinentes: máximo Dimensión: `PipelineName`
`s3.dlqS3RecordsSuccess.count`	El número de registros que se enviaron correctamente a la cola de mensajes fallidos de S3. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`s3.dlqS3RecordsFailed.count`	El número de registros que no se pudieron enviar a la cola de mensajes fallidos de S3. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`s3.dlqS3RequestSuccess.count`	El número de solicitudes que se enviaron correctamente a la cola de mensajes fallidos de S3. Estadísticas pertinentes: suma Dimensión: `PipelineName`

Sufijo de la métrica	Descripción
`s3.dlqS3RequestFailed.count`	El número de solicitudes fallidas que se enviaron a la cola de mensajes fallidos de S3. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`s3.dlqS3RequestLatency.count`	Un recuento de los puntos de datos registrados mientras se envían las solicitudes a la cola de mensajes fallidos de S3, incluidos los reintentos. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`s3.dlqS3RequestLatency.sum`	La latencia total de las solicitudes enviadas a la cola de mensajes fallidos de S3, incluidos los reintentos, en milisegundos. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`s3.dlqS3RequestLatency.max`	La latencia máxima de las solicitudes enviadas a la cola de mensajes fallidos de S3, incluidos los reintentos, en milisegundos. Estadísticas pertinentes: máximo Dimensión: `PipelineName`
`s3.dlqS3RequestSizeBytes.count`	Un recuento de la distribución de los tamaños de carga útil de las solicitudes a la cola de mensajes fallidos de S3, en bytes. Estadísticas pertinentes: suma Dimensión: `PipelineName`

Sufijo de la métrica	Descripción
`s3.dlqS3RequestSizeBytes.sum`	La distribución total de los tamaños de carga útil de las solicitudes a la cola de mensajes fallidos de S3, en bytes. Estadísticas pertinentes: suma Dimensión: `PipelineName`
`s3.dlqS3RequestSizeBytes.max`	La distribución máxima de los tamaños de carga útil de las solicitudes a la cola de mensajes fallidos de S3, en bytes. Estadísticas pertinentes: máximo Dimensión: `PipelineName`

Métricas del sistema y de medición

Las siguientes métricas de se aplican al sistema general de OpenSearch Ingestion. Estas métricas no llevan ningún prefijo.

Métrica	Descripción
`system.cpu.usage.value`	El porcentaje de uso de CPU disponible para todos los nodos de datos. Estadísticas pertinentes: promedio Dimensión: `PipelineName` , `area`, `id`
`system.cpu.count.value`	La cantidad total de uso de la CPU para todos los nodos de datos. Estadísticas pertinentes: promedio Dimensión: `PipelineName` , `area`, `id`
`jvm.memory.max.value`	La cantidad máxima de memoria que se puede usar para la administración de la memoria, en bytes.

Métrica	Descripción
	Estadísticas pertinentes: promedio Dimensión: `PipelineName` , `area`, `id`
`jvm.memory.used.value`	La cantidad total de memoria usada, en bytes. Estadísticas pertinentes: promedio Dimensión: `PipelineName` , `area`, `idsigna`
`jvm.memory.committed.value`	La cantidad de memoria asignada para el uso de la máquina virtual Java (JVM), en bytes. Estadísticas pertinentes: promedio Dimensión: `PipelineName` , `area`, `id`
`computeUnits`	El número de unidades de cómputo de OpenSearch Ingestion (OCU de ingestión) que utiliza una canalización. Estadísticas pertinentes: máximo, suma, promedio Dimensión: `PipelineName`

Prácticas recomendadas para Amazon OpenSearch Ingestion

En este tema se explican las prácticas recomendadas sobre la creación y administración de canalizaciones de Amazon OpenSearch Ingestion e incluye instrucciones generales que se aplican a muchos casos de uso. Cada carga de trabajo es única, con características únicas, por lo que ninguna recomendación genérica es exactamente adecuada para cada caso de uso.

Temas

- Prácticas recomendadas generales
- Alarmas de CloudWatch recomendadas

Prácticas recomendadas generales

Las siguientes prácticas recomendadas generales se aplican a la creación y administración de canalizaciones.

- Para garantizar una alta disponibilidad, configure las canalizaciones de VPC con dos o tres subredes. Si solo implementa una canalización en una subred y la zona de disponibilidad deja de funcionar, no podrá incorporar datos.
- Dentro de cada canalización, recomendamos limitar la cantidad de subcanalizaciones a 5 o menos.
- Si utiliza el complemento de fuente de S3, utilice archivos S3 de tamaño uniforme para obtener un rendimiento óptimo.
- Si utiliza el complemento de fuente de S3, añada 30 segundos de tiempo de espera de visibilidad adicional por cada 0,25 GB de tamaño de archivo del bucket de S3 para obtener un rendimiento óptimo.
- Incluya una cola de mensajes fallidos (DLQ) en la configuración de canalización para poder descargar los eventos fallidos y hacerlos accesibles para su análisis. Si los receptores rechazan los datos debido a mapeos incorrectos u otros problemas, puede enrutar los datos al DLQ para solucionar el problema.

Alarmas de CloudWatch recomendadas

Las alarmas de CloudWatch realizan una acción cuando una métrica de CloudWatch supera un valor especificado para un periodo de tiempo determinado. Por ejemplo, es posible que desee que AWS envíe un email si el estado del clúster es `red` durante más de un minuto. En esta sección se incluyen algunas alarmas recomendadas para Amazon OpenSearch Ingestion y cómo responder a ellas.

Para obtener más información sobre la configuración de las alarmas, consulte Creación de alarmas de Amazon CloudWatch en la Guía del usuario de Amazon CloudWatch.

Alarma	Problema
`computeUnits` máximas es = las `maxUnits` configuradas durante 15	La canalización ha alcanzado su capacidad máxima y es posible que necesite una actualización de `maxUnits`. Aumente la capacidad máxima de su canalización

Alarma	Problema
minutos, 3 veces consecutivas	
Suma de `opensearch.documentErrors.count` es = suma de *`{sub_pipeline_name}`*`.opensearch.recordsIn.count` durante 1 minuto, 1 periodo consecutivo	La canalización no puede escribir en el receptor de OpenSearch. Compruebe los permisos de la canalización y confirme que el dominio o la colección están en buen estado. También puede comprobar si hay eventos fallidos en la cola de mensajes fallidos (DLQ), si está configurada.
`bulkRequestLatency.max` máximo es >= x durante 1 minuto, 1 periodo consecutivo	La canalización está experimentando una latencia alta al enviar datos al receptor de OpenSearch. Es probable que esto se deba a que el receptor tiene un tamaño insuficiente o a una mala estrategia de partición, lo que provoca que el receptor quede rezagado. Una latencia alta y sostenida puede afectar el rendimiento de la canalización y es probable que genere una contrapresión para los clientes.
Suma de `httpAuthFailure.count` >= 1 durante 1 minuto, 1 periodo consecutivo	Las solicitudes de ingesta no se autentican. Confirme que todos los clientes tengan habilitada correctamente la autenticación de Signature Version 4.
`system.cpu.usage.value` promedio es >= 80 % durante 15 minutos, 3 periodos consecutivos	Un uso elevado y sostenido de la CPU puede resultar problemático. Considere aumentar la capacidad máxima de la canalización.

Alarma	Problema
`bufferUsage.value` promedio es >= 80 % durante 15 minutos, 3 periodos consecutivos	Un uso elevado y sostenido del búfer puede resultar problemático. Considere aumentar la capacidad máxima de la canalización.

Otras alarmas para tener en cuenta

Considere configurar las siguientes alarmas en función de las características de Amazon OpenSearch Ingestion que utilice habitualmente.

Alarma	Problema
`dynamodb.exportJobFailure.count` suma 1	Falló el intento de activar una exportación a Amazon S3.
El promedio de `opensearch.EndtoEndLatency.avg` es > X durante 15 minutos, 4 periodos consecutivos	La `EndtoEndLatency` es superior a la deseada para la lectura de las transmisiones de DynamoDB. Esto puede deberse a un clúster de OpenSearch con una escala insuficiente o a una capacidad máxima de OCU de canalización demasiado baja para el rendimiento de la WCU de la tabla de DynamoDB. La `EndtoEndLatency` será mayor después de una exportación, pero debería disminuir con el tiempo a medida que se vaya actualizando con las últimas transmisiones de DynamoDB.
`dynamodb.changeEventsProcessed.count` suma == 0 durante X minutos	No se está recopilando ningún registro de las transmisiones de DynamoDB. Esto puede deberse a una falta de actividad en la tabla o a un problema al acceder a las transmisiones de DynamoDB.

Alarma	Problema
Suma de `opensearch.s3.dlqS3RecordsSuccess.count` >= suma de `opensearch.documentSuccess.count` durante 1 minuto, 1 periodo consecutivo	Se envía un número mayor de registros al DLQ que al receptor de OpenSearch. Revise las métricas del complemento del receptor de OpenSearch para investigar y determinar la causa raíz.
Suma de `grok.grokProcessingTimeouts.count` = suma de recordsIn.count durante 1 minuto, 5 períodos consecutivos	Se agota el tiempo de espera de todos los datos mientras el procesador Grok intenta hacer coincidir el patrón. Es probable que esto afecte el rendimiento y ralentice la canalización. Considere ajustar los patrones para reducir los tiempos de espera.
Suma de `grok.grokProcessingErrors.count` es >= 1 durante 1 minuto, 1 periodo consecutivo	El procesador Grok no consigue hacer coincidir los patrones con los datos de la canalización, lo que provoca errores. Revise los datos y las configuraciones del complemento de Grok para asegurarse de que se espera que coincidan los patrones.
Suma de `grok.grokProcessingMismatch.count` = suma de recordsIn.count durante 1 minuto, 5 periodos consecutivos	El procesador Grok no puede hacer coincidir los patrones con los datos de la canalización. Revise los datos y las configuraciones del complemento de Grok para asegurarse de que se espera que coincidan los patrones.

Alarma	Problema
Suma de `date.dateProcessingMatchFailure.count` = suma de recordsIn.count durante 1 minuto, 5 periodos consecutivos	El procesador Date no puede hacer coincidir ningún patrón con los datos de la canalización. Revise los datos y las configuraciones del complemento de Date para asegurarse de que se espera el patrón.
Suma de `s3.s3ObjectsFailed.count` >= 1 durante 1 minuto, 1 periodo consecutivo	Este problema se debe a que el objeto S3 no existe o a que la canalización no tiene privilegios suficientes. Revise las métricas `s3ObjectsNotFound.count` y `s3ObjectsAccessDenied.count` para determinar la causa raíz. Confirme que el objeto S3 existe o actualice los permisos.
Suma de `s3.sqsMessagesFailed.count` >= 1 durante 1 minuto, 1 periodo consecutivo	El complemento S3 no pudo procesar un mensaje de Amazon SQS. Si tiene una DLQ activada en la cola de SQS, revise el mensaje de error. Es posible que la cola esté recibiendo datos no válidos que la canalización está intentando procesar.
Suma de `http.badRequests.count` >= 1 durante 1 minuto, 1 periodo consecutivo	El cliente está enviando una solicitud errónea. Confirme que todos los clientes están enviando la carga útil adecuada.
Suma de `http.requestsTooLarge.count` >= 1 durante 1 minuto, 1 periodo consecutivo	Las solicitudes del complemento de fuente HTTP contienen demasiados datos, lo que supera la capacidad del búfer. Ajuste el tamaño del lote para sus clientes.

Alarma	Problema
Suma de `http.internalServerError.count` >= 0 durante 1 minuto, 1 periodo consecutivo	El complemento de fuente HTTP tiene problemas para recibir eventos.
Suma de `http.requestTimeouts.count` >= 0 durante 1 minuto, 1 periodo consecutivo	Es probable que los tiempos de espera de la fuente se deban a que la canalización está mal aprovisionada. Considere aumentar las `maxUnits` de la canalización para gestionar la carga de trabajo adicional.
Suma de `otel_trace.badRequests.count` >= 1 durante 1 minuto, 1 periodo consecutivo	El cliente está enviando una solicitud errónea. Confirme que todos los clientes están enviando la carga útil adecuada.
Suma de `otel_trace.requestsTooLarge.count` >= 1 durante 1 minuto, 1 periodo consecutivo	Las solicitudes del complemento de fuente de OTel contienen demasiados datos, lo que supera la capacidad del búfer. Ajuste el tamaño del lote para sus clientes.
Suma de `otel_trace.internalServerError.count` >= 0 durante 1 minuto, 1 periodo consecutivo	El complemento de fuente Otel Trace tiene problemas para recibir eventos.

Alarma	Problema
Suma de `otel_trace.requestTimeouts.count` >= 0 durante 1 minuto, 1 periodo consecutivo	Es probable que los tiempos de espera de la fuente se deban a que la canalización está mal aprovisionada. Considere aumentar las `maxUnits` de la canalización para gestionar la carga de trabajo adicional.
Suma de `otel_metrics.requestTimeouts.count` >= 0 durante 1 minuto, 1 periodo consecutivo	Es probable que los tiempos de espera de la fuente se deban a que la canalización está mal aprovisionada. Considere aumentar las `maxUnits` de la canalización para gestionar la carga de trabajo adicional.

Amazon OpenSearch sin servidor

Amazon OpenSearch sin servidor es una configuración de escalado automático bajo demanda para Amazon OpenSearch Service. A diferencia de los dominios aprovisionados de OpenSearch, que requieren una administración manual de la capacidad, una colección de OpenSearch sin servidor escala automáticamente los recursos de cómputo según las necesidades de su aplicación.

OpenSearch sin servidor ofrece una solución rentable para cargas de trabajo poco frecuentes, intermitentes o impredecibles. Optimiza los costos al escalar automáticamente la capacidad de cómputo según el uso de su aplicación. Las colecciones sin servidor utilizan el mismo volumen de almacenamiento distribuido, de gran capacidad y alta disponibilidad que los dominios aprovisionados de Amazon OpenSearch Service.

Las colecciones de OpenSearch sin servidor siempre están cifradas. Puede elegir la clave de cifrado, pero no puede desactivar el cifrado. Para obtener más información, consulte the section called “Cifrado”

Ventajas

OpenSearch sin servidor tiene los siguientes beneficios:

- Más simple que el aprovisionado: OpenSearch sin servidor elimina gran parte de la complejidad de administrar los clústeres y la capacidad de OpenSearch. Ajusta y dimensiona de forma automática los clústeres, y se encarga de la administración del ciclo de vida de las particiones y los índices. También gestiona las actualizaciones del software del servicio y de las versiones de OpenSearch. Todas las actualizaciones y mejoras no son disruptivas.
- Rentable: cuando utiliza OpenSearch sin servidor, solo paga por los recursos que consume. Esto elimina la necesidad del aprovisionamiento inicial y del aprovisionamiento excesivo para las cargas de trabajo máximas.
- Alta disponibilidad: OpenSearch sin servidor admite las cargas de trabajo de producción con redundancia para proteger contra las interrupciones de la zona de disponibilidad y los fallos de la infraestructura.
- Escalable: OpenSearch sin servidor escala de forma automática los recursos para mantener una rapidez constante en las tasas de ingesta de datos y los tiempos de respuesta a las consultas.

¿Qué es Amazon OpenSearch sin servidor?

Amazon OpenSearch sin servidor es una opción bajo demanda y sin servidor para Amazon OpenSearch Service que elimina la complejidad operativa de aprovisionar, configurar y ajustar clústeres de OpenSearch. Es ideal para organizaciones que prefieren no administrar sus propios clústeres o que no cuentan con los recursos y el conocimiento experto necesarios para operar implementaciones a gran escala. Con OpenSearch sin servidor, usted puede buscar y analizar grandes volúmenes de datos sin administrar la infraestructura subyacente.

Una colección de OpenSearch sin servidor es un grupo de índices de OpenSearch que funcionan juntos para soportar una carga de trabajo o un caso de uso específicos. Las colecciones simplifican las operaciones en comparación con los clústeres autoadministrados de OpenSearch, que requieren aprovisionamiento manual.

Las colecciones utilizan el mismo almacenamiento distribuido, de gran capacidad y alta disponibilidad que los dominios aprovisionados de Amazon OpenSearch Service, pero reducen aún más la complejidad al eliminar la configuración y el ajuste manual. Los datos dentro de una colección están cifrados en tránsito. OpenSearch sin servidor también es compatible con OpenSearch Dashboards, lo que proporciona una interfaz para el análisis de datos.

Actualmente, las colecciones sin servidor ejecutan la versión 2.17.x de OpenSearch. A medida que se publican nuevas versiones, OpenSearch sin servidor actualiza automáticamente las colecciones para incorporar nuevas características, correcciones de errores y mejoras de rendimiento.

OpenSearch sin servidor admite las mismas operaciones de la API de ingesta y consulta que el paquete de código abierto de OpenSearch, por lo que puede seguir utilizando sus clientes y aplicaciones existentes. Sus clientes deben ser compatibles con OpenSearch 2.x para poder trabajar con OpenSearch sin servidor. Para obtener más información, consulte the section called “Ingesta de datos en las colecciones”.

Temas

- Casos de uso de OpenSearch sin servidor
- Funcionamiento
- Elección de un tipo de colección
- Precios
- Regiones de AWS compatible
- Limitaciones

- Comparación entre OpenSearch Service y OpenSearch sin servidor

Casos de uso de OpenSearch sin servidor

OpenSearch sin servidor admite dos casos de uso principales:

- Análisis de registros: el segmento de análisis de registros se centra en analizar grandes volúmenes de datos de series temporales semiestructurados y generados por máquinas para obtener información operativa y sobre el comportamiento de los usuarios.
- Búsqueda de texto completo: el segmento de búsqueda de texto completo potencia las aplicaciones de sus redes internas (sistemas de administración de contenido, documentos legales) y las aplicaciones orientadas a Internet, como la búsqueda de contenido de sitios web de comercio electrónico.

Al crear una colección, debe elegir uno de estos casos de uso. Para obtener más información, consulte the section called “Elección de un tipo de colección”.

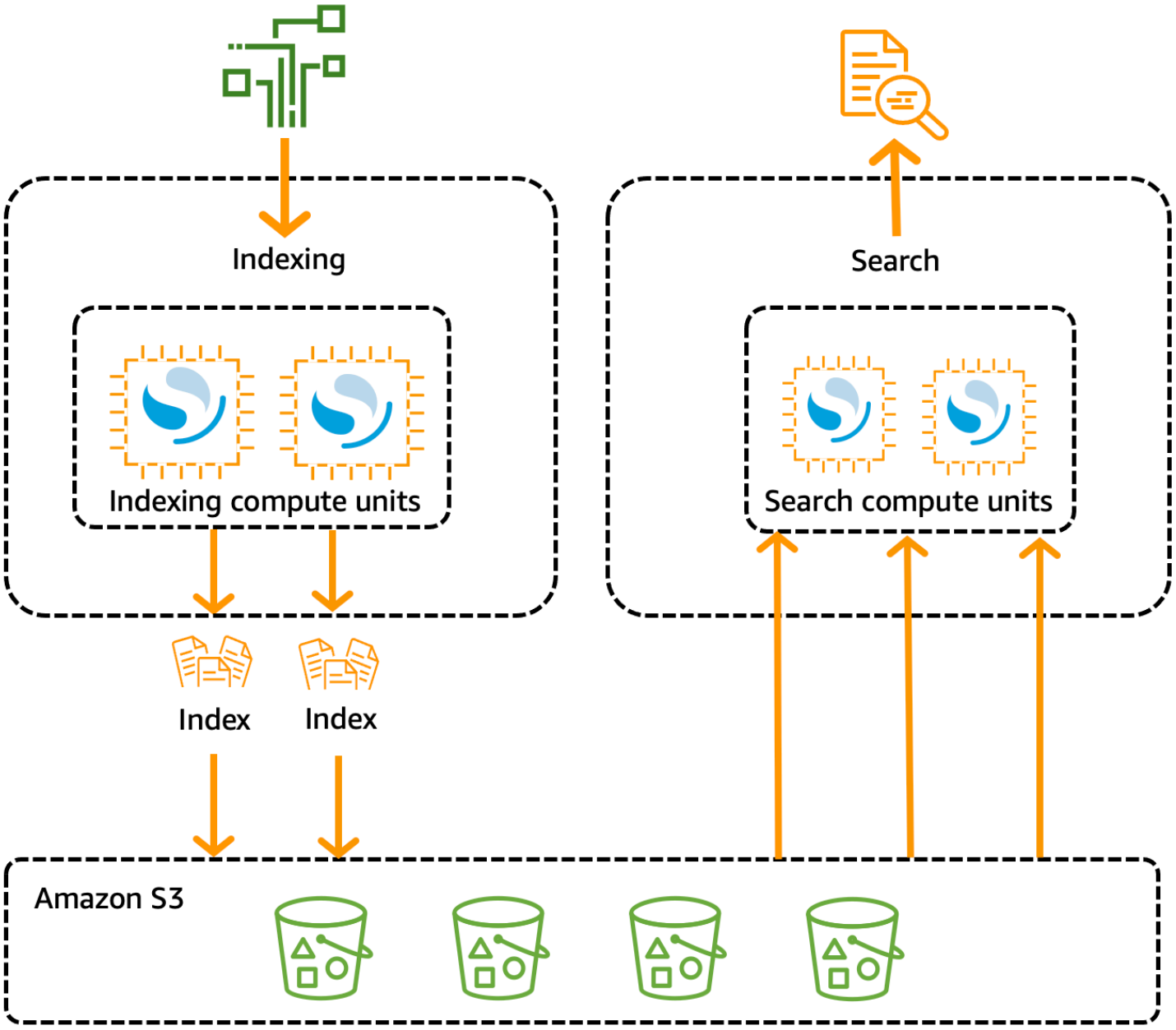
Funcionamiento

Los clústeres tradicionales de OpenSearch tienen un único conjunto de instancias que realizan operaciones de indexación y búsqueda, y el almacenamiento de índices está vinculado de forma estrecha con la capacidad de cómputo. Por el contrario, OpenSearch sin servidor utiliza una arquitectura nativa en la nube que separa los componentes de indexación (ingesta) de los componentes de búsqueda (consulta), con Amazon S3 como el almacenamiento de datos principal para los índices.

Esta arquitectura desacoplada permite escalar las funciones de búsqueda e indexación de forma independiente entre ellas y de los datos indexados en S3. La arquitectura también proporciona aislamiento para las operaciones de ingesta y consulta, de modo que puedan ejecutarse de forma simultánea sin contención de recursos.

Cuando escribe datos en una colección, OpenSearch sin servidor los distribuye a las unidades de computación de indexación. Las unidades de computación de indexación ingieren los datos entrantes y mueven los índices a S3. Al realizar una búsqueda en los datos de la colección, OpenSearch sin servidor dirige las solicitudes a las unidades de computación de búsqueda que contienen los datos consultados. Las unidades de computación de búsqueda descargan los datos indexados de forma directa desde S3 (si aún no están almacenados en la memoria caché local), ejecutan operaciones de búsqueda y realizan agregaciones.

La siguiente imagen ilustra esta arquitectura desacoplada:



La capacidad de computación de OpenSearch sin servidor para la ingestión, la búsqueda y la consulta de datos se mide en unidades de computación de OpenSearch (OCU). Cada OCU es una combinación de 6 GiB de memoria y la CPU virtual (vCPU) correspondiente, así como la transferencia de datos a Amazon S3. Cada OCU incluye suficiente almacenamiento efímero en caliente para 120 GiB de datos de índice.

Al crear su primera colección, OpenSearch sin servidor crea una instancia de dos OCU: una para la indexación y otra para la búsqueda. Para garantizar la alta disponibilidad, también lanza un conjunto de nodos en espera en otra zona de disponibilidad. Para fines de desarrollo y pruebas, puede

deshabilitar la configuración Habilitar redundancia para una colección, lo que elimina las dos réplicas en espera y solo crea una instancia de dos OCU. De forma predeterminada, las réplicas activas redundantes están habilitadas, lo que significa que se crean instancias para un total de cuatro OCU para la primera colección de una cuenta.

Estas OCU existen incluso cuando no hay actividad en ningún punto de conexión de la colección. Todas las colecciones posteriores comparten estas OCU. Al crear más colecciones en la misma cuenta, OpenSearch sin servidor solo agrega OCU adicionales para buscar e ingerir según sea necesario para respaldar las colecciones, de acuerdo con los límites de capacidad que especifique. La capacidad no se reduce verticalmente a medida que disminuye el uso de computación.

Para obtener más información sobre cómo se facturan estas OCU, consulte the section called "Precios".

Elección de un tipo de colección

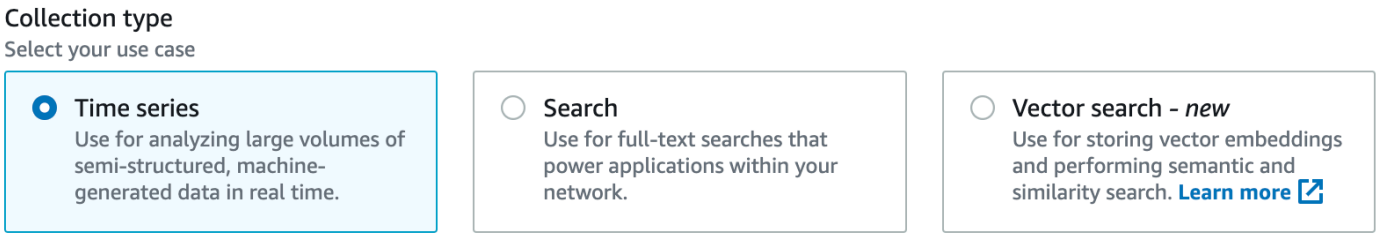
OpenSearch sin servidor admite tres tipos principales de colecciones:

Series temporales: el segmento de análisis de registros que analiza grandes volúmenes de datos semiestructurados generados por máquinas en tiempo real y que proporciona información sobre operaciones, seguridad, comportamiento de usuarios y rendimiento del negocio.

Búsqueda: búsqueda de texto completo que habilita aplicaciones dentro de redes internas, como sistemas de administración de contenido y repositorios de documentos legales, así como aplicaciones orientadas a internet, como búsquedas en sitios de comercio electrónico y detección de contenido.

Búsqueda vectorial: la búsqueda semántica basada en incrustaciones vectoriales simplifica la administración de datos vectoriales y habilita experiencias de búsqueda mejoradas con machine learning (ML). Es compatible con aplicaciones de IA generativa, como chatbots, asistentes personales y detección de fraude.

El tipo de colección se elige cuando se crea una colección por primera vez:



El tipo de colección que elija dependerá del tipo de datos que piensa incorporar a la colección y de cómo piensa consultarlos. No puede cambiar el tipo de colección después de crearla.

Los tipos de colecciones presentan las siguientes diferencias notables:

- En el caso de las colecciones de búsqueda y de búsqueda vectorial, todos los datos se almacenan en un almacenamiento en caliente para garantizar tiempos de respuesta rápidos a las consultas. Las colecciones de series temporales utilizan una combinación de almacenamiento en caliente y templado, donde los datos más recientes se guardan en un almacenamiento en caliente para optimizar los tiempos de respuesta a las consultas para los datos a los que se accede con más frecuencia.
- En el caso de las colecciones de series temporales y de búsqueda vectorial, no puede indexar por identificador de documento personalizado ni actualizarlas mediante solicitudes indirectas. Esta operación se reserva para los casos de uso de búsqueda. En su lugar, puede actualizar por ID de documento. Para obtener más información, consulte the section called “Permisos y operaciones de la API de OpenSearch”.
- Para las recopilaciones de series temporales y de búsqueda, no puede utilizar índices de tipo k-NN.

Precios

AWS le cobra por los siguientes componentes de OpenSearch sin servidor:

- Computación de la ingesta de datos
- Computación de búsquedas y consultas
- Almacenamiento retenido en Amazon S3

Una OCU comprende 6 GB de RAM, vCPU correspondiente, almacenamiento GP3 y transferencia de datos hacia Amazon S3. La unidad mínima que se puede facturar es de 0,5 OCU. AWS factura las OCU por hora, con granularidad por segundo. En su estado de cuenta, usted ve una entrada por cómputo en OCU-horas con una etiqueta ingesta de datos y una etiqueta para búsqueda. AWS también factura mensualmente por los datos almacenados en Amazon S3. No le cobra por usar OpenSearch Dashboards.

Cuando usted crea una colección con réplicas activas redundantes, se le factura un mínimo de 2 OCU:

- 1 OCU (0,5 OCU × 2) para ingesta, incluye tanto la primaria como la de respaldo
- 1 OCU (0,5 OCU × 2) para la búsqueda

Si usted deshabilita las réplicas activas redundantes, se le factura un mínimo de 1 OCU (0,5 OCU × 2) para la primera colección en su cuenta. Todas las colecciones posteriores pueden compartir esas OCU.

OpenSearch sin servidor agrega OCU adicionales en incrementos de 1 OCU en función del almacenamiento y la capacidad de computación necesarios para respaldar sus colecciones. Puede configurar un número máximo de OCU para su cuenta con el fin de controlar los costos.

> **Note**
>
> Las colecciones con AWS KMS keys únicas no pueden compartir las OCU con otras colecciones.

OpenSearch sin servidor intenta utilizar los recursos mínimos necesarios para tener en cuenta los cambios en las cargas de trabajo. La cantidad de OCU aprovisionadas en un momento dado puede variar y no es exacta. Con el tiempo, el algoritmo que utiliza OpenSearch sin servidor seguirá mejorando para minimizar el uso del sistema.

Para obtener información completa sobre precios, consulte los precios de Amazon OpenSearch Service.

Regiones de AWS compatible

OpenSearch sin servidor está disponible en un subconjunto de Regiones de AWS en las que está disponible OpenSearch Service. Para obtener una lista de las regiones compatibles, consulte Puntos de conexión y cuotas de Amazon OpenSearch Service en Referencia general de AWS.

Limitaciones

OpenSearch sin servidor tiene las siguientes limitaciones:

- No se admiten algunas operaciones de la API de OpenSearch. Consulte the section called “Permisos y operaciones de la API de OpenSearch”.
- No se admiten algunos complementos de OpenSearch. Consulte the section called “Complementos de OpenSearch compatibles”.

- En la actualidad, no hay forma de migrar de forma automática los datos desde un dominio administrado por OpenSearch Service a una colección sin servidor. Debe volver a indexar los datos desde un dominio a una colección.
- No se admiten el acceso entre cuentas a las colecciones. No puede incluir colecciones de otras cuentas en las políticas de cifrado o de acceso a los datos.
- No se admiten los complementos personalizados de OpenSearch.
- No puede tomar ni restaurar instantáneas de las colecciones de OpenSearch sin servidor.
- No se admiten la búsqueda y la replicación entre regiones.
- Hay límites para la cantidad de recursos sin servidor que puede tener en una sola cuenta y región. Consulte las Cuotas de OpenSearch sin servidor.
- El intervalo de actualización de los índices de las colecciones de búsqueda vectorial es de aproximadamente 60 segundos. El intervalo de actualización de los índices de las colecciones de búsqueda y serie temporal es de aproximadamente 10 segundos.
- El número de particiones, número de intervalos e intervalo de actualización no se pueden modificar y son gestionados por OpenSearch sin servidor. La estrategia de partición se basa en el tipo de colección y el tráfico. Por ejemplo, una colección de series temporales escala las particiones principales en función de los cuellos de botella del tráfico de escritura.
- Se admiten las características geoespaciales disponibles en las versiones de OpenSearch hasta la 2.1.

Comparación entre OpenSearch Service y OpenSearch sin servidor

En OpenSearch sin servidor, algunos conceptos y características son diferentes de su característica correspondiente para un dominio de OpenSearch Service aprovisionado. Por ejemplo, una diferencia importante es que OpenSearch sin servidor no tiene el concepto de clúster o nodo.

La siguiente tabla describe cómo las características y los conceptos importantes de OpenSearch sin servidor se diferencian de las características equivalentes en un dominio OpenSearch Service aprovisionado.

Característica	OpenSearch Service	OpenSearch sin servidor
Dominios frente a colecciones	Los índices se encuentran en dominios, que son clústeres	Los índices se encuentran en colecciones, que son agrupaciones lógicas de índices que

Característica	OpenSearch Service	OpenSearch sin servidor
	de OpenSearch que ya están aprovisionados. Para obtener más información, consulte Creación y administración de dominios.	representan una carga de trabajo o un caso de uso específicos. Para obtener más información, consulte the section called “Administración de colecciones”.
Tipos de nodos y administración de la capacidad	Crea un clúster con tipos de nodos que cumplen con sus especificaciones de costo y rendimiento. Debe calcular sus propios requisitos de almacenamiento y elegir un tipo de instancia para su dominio. Para obtener más información, consulte the section called “Determinación del tamaño de dominios”.	OpenSearch sin servidor escala y aprovisiona de forma automática unidades de computación adicionales para su cuenta en función del uso de la capacidad. Para obtener más información, consulte the section called “Administración de los límites de capacidad”.
Facturación de	Paga por cada hora de uso de una instancia EC2 y por el tamaño acumulado de cualquier volumen de almacenamiento de EBS adjunto a sus instancias. Para obtener más información, consulte the section called “Precios”.	Se le cobrará en horas de OCU por la computación para la ingesta de datos, la computación para las búsquedas y las consultas y el almacenamiento retenido en S3. Para obtener más información, consulte the section called “Precios”.

Característica	OpenSearch Service	OpenSearch sin servidor
Cifrado	El cifrado en reposo es opcional para los dominios. Para obtener más informaci ón, consulte the section called “Cifrado en reposo”.	El cifrado en reposo es obligatorio para las colecciones. Para obtener más información, consulte the section called “Cifrado”.
Control de acceso a los datos	El acceso a los datos dentro de los dominios se determina mediante políticas de IAM y un control de acceso detallado.	El acceso a los datos dentro de las colecciones se determina mediante las políticas de acceso a los datos.
Operacion es de OpenSearch compatibles	OpenSearch Service admite un subconjunto de todas las operaciones de la API de OpenSearch. Para obtener más informaci ón, consulte the section called “Operaciones admitidas”.	OpenSearch sin servidor admite un subconjun to diferente de operaciones de la API de OpenSearch. Para obtener más información, consulte the section called “Operaciones y complementos compatibles”.
Inicio de sesión en Dashboards	Inicio de sesión con un nombre de usuario y una contraseña. Para obtener más informaci ón, consulte the section called “Acceder a OpenSearch Dashboards como usuario maestro”.	Si ha iniciado sesión en la consola AWS y accede a la URL de su panel de control, iniciará sesión automáticamente. Para obtener más información, consulte the section called “Acceso a OpenSearch Dashboards”.
API	Interactúe mediante la programación con OpenSearch Service con las operaciones de la API de OpenSearch Service.	Interactúe mediante la programación con OpenSearch sin servidor con las operaciones de la API de OpenSearch sin servidor.

Característica	OpenSearch Service	OpenSearch sin servidor
Acceso a la red	La configuración de red de un dominio se aplica tanto al punto de conexión del dominio como al punto de conexión de OpenSearch Dashboard s. El acceso a la red para ambos está vinculado de forma estrecha.	La configuración de red para el punto de conexión del dominio y el punto de conexión de OpenSearch Dashboards está desacopla da. Puede optar por no configurar el acceso a la red para OpenSearch Dashboards. Para obtener más información, consulte the section called “Acceso a la red”.
Firma de solicitudes	Utilice los clientes REST de alto y bajo nivel de OpenSearc h para firmar las solicitudes. Especifique el nombre del servicio como `es`.	En este momento, OpenSearch sin servidor admite un subconjunto de clientes que admite OpenSearch Service. Al firmar las solicitudes, especifique el nombre del servicio como `aoss`. El encabezado `x-amz-content-sha256` es obligatorio. Para obtener más información, consulte the section called “Otros clientes”.
Actualiza ciones de la versión de OpenSearch	Actualice sus dominios de forma manual a medida que aparezcan nuevas versiones de OpenSearch disponibles. Es responsable de garantiza r que su dominio cumpla con los requisitos de actualización y de que solucionar cualquier cambio importante.	OpenSearch sin servidor actualiza de forma automática sus colecciones a las nuevas versiones de OpenSearch. Las actualiza ciones no siempre se producen en cuanto está disponible una nueva versión.
Actualiza ciones del software del servicio	De forma manual, aplique las actualizaciones del software del servicio a su dominio a medida que estén disponibles.	OpenSearch sin servidor actualiza de forma automática sus colecciones para incluir las últimas correcciones de errores, características y mejoras de rendimiento.

Característica	OpenSearch Service	OpenSearch sin servidor
Acceso mediante VPC	Puede aprovisionar su dominio dentro de una VPC. También puede crear puntos de conexión de VPC administrados por OpenSearch Service adicionales para acceder al dominio.	Cree un punto de conexión de VPC administrado por OpenSearch sin servidor o más para su cuenta. A continuación, incluya estos puntos de conexión dentro de las políticas de red.
Autenticación SAML	La autenticación SAML se habilita por dominio. Para obtener más información, consulte the section called “Autenticación SAML para OpenSearch Dashboards”.	Configure uno o más proveedores de SAML a nivel de cuenta y, a continuación, incluya los ID de usuario y de grupo asociados dentro de las políticas de acceso a los datos. Para obtener más información, consulte the section called “Autenticación SAML”.
seguridad de la capa de transporte (TLS)	OpenSearch Service admite TLS 1.2, pero se recomienda usar TLS 1.3.	OpenSearch sin servidor admite TLS 1.2, pero se recomienda usar TLS 1.3.

Tutorial: introducción a Amazon OpenSearch sin servidor

El tutorial muestra los pasos básicos para poner en marcha rápidamente una colección de búsqueda de Amazon OpenSearch sin servidor. Una colección de búsqueda le permite potenciar las aplicaciones de sus redes internas y las aplicaciones orientadas a Internet, como la búsqueda en sitios web de comercio electrónico y la búsqueda de contenido.

Para obtener información sobre cómo utilizar una colección de búsqueda vectorial, consulte. the section called “Trabajo con colecciones de búsqueda vectorial” Para obtener información detallada sobre el uso de colecciones, consulte the section called “Administración de colecciones” y los demás temas de esta guía.

En este tutorial, deberá completar los siguientes pasos:

1. Configurar permisos
2. Crear una colección
3. Cargar y buscar datos
4. Eliminar la colección

> **Note**
>
> Recomendamos que usted utilice únicamente caracteres ASCII para su `IndexName`. Si no utiliza caracteres ASCII para `IndexName`, `IndexName` en las métricas de CloudWatch se convertirá a un formato codificado en URL para caracteres que no sean ASCII.

Paso 1: configurar permisos

Para completar este tutorial y utilizar OpenSearch sin servidor en general, debe tener los permisos de IAM correctos. En este tutorial, creará una colección, cargará y buscará datos y, a continuación, eliminará la colección.

Su usuario o rol debe tener adjunta una política basada en la identidad con los siguientes permisos mínimos:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Action": [
        "aoss:CreateCollection",
        "aoss:ListCollections",
        "aoss:BatchGetCollection",
        "aoss:DeleteCollection",
        "aoss:CreateAccessPolicy",
        "aoss:ListAccessPolicies",
        "aoss:UpdateAccessPolicy",
        "aoss:CreateSecurityPolicy",
        "aoss:GetSecurityPolicy",
        "aoss:UpdateSecurityPolicy",
        "iam:ListUsers",
```

```
        "iam:ListRoles"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

Para obtener más información sobre los permisos de IAM de OpenSearch sin servidor, consulte the section called “Identity and Access Management”

Paso 2: crear una colección

Una colección es un grupo de índices de OpenSearch que funcionan juntos para soportar una carga de trabajo o un caso de uso específicos.

Para crear una colección de OpenSearch sin servidor

1. Abra la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. Seleccione Colecciones en el panel de navegación de la izquierda y elija Crear colección.
3. Asigne un nombre a las películas de la colección.
4. Para el tipo de colección, seleccione Buscar. Para obtener más información, consulte Choosing a collection type (Elección de un tipo de recopilación).
5. En Seguridad, elija Creación estándar.
6. En Cifrado, seleccione Utilizar Clave propiedad de AWS. Esta es la AWS KMS key que va a utilizar OpenSearch sin servidor para cifrar los datos.
7. En Red, configure los ajustes de red para la colección.
   - Para el tipo de acceso, seleccione Público.
   - Para el tipo de recurso, elija tanto Habilitar el acceso a puntos de conexión de OpenSearch como Habilitar el acceso a OpenSearch Dashboards. Dado que cargará y buscará datos mediante OpenSearch Dashboards, deberá habilitar ambos.
8. Seleccione Siguiente.
9. En Configurar el acceso a los datos, defina los ajustes de acceso a la colección. Las políticas de acceso a los datos le permiten a los usuarios y a los roles acceder a los datos de una colección. En este tutorial, le brindaremos a un solo usuario los permisos necesarios para indexar y buscar datos en la colección de películas.

Cree una sola regla que proporcione acceso a la colección películas. Asígnele a la regla el nombre Acceso a la colección de películas.

10. Seleccione Agregar entidades principales, Usuarios y roles de IAM, y luego el usuario de IAM que utilizará para iniciar sesión en OpenSearch Dashboards e indexar datos. Seleccione Guardar.
11. En Indexar permisos, seleccione todos los permisos.
12. Seleccione Siguiente.
13. En la configuración de la política de acceso, seleccione Crear una nueva política de acceso a datos y póngale el nombre películas a la política.
14. Seleccione Siguiente.
15. Revise la configuración de la colección y seleccione Enviar. Espere unos minutos hasta que el estado de la colección cambie a `Active`.

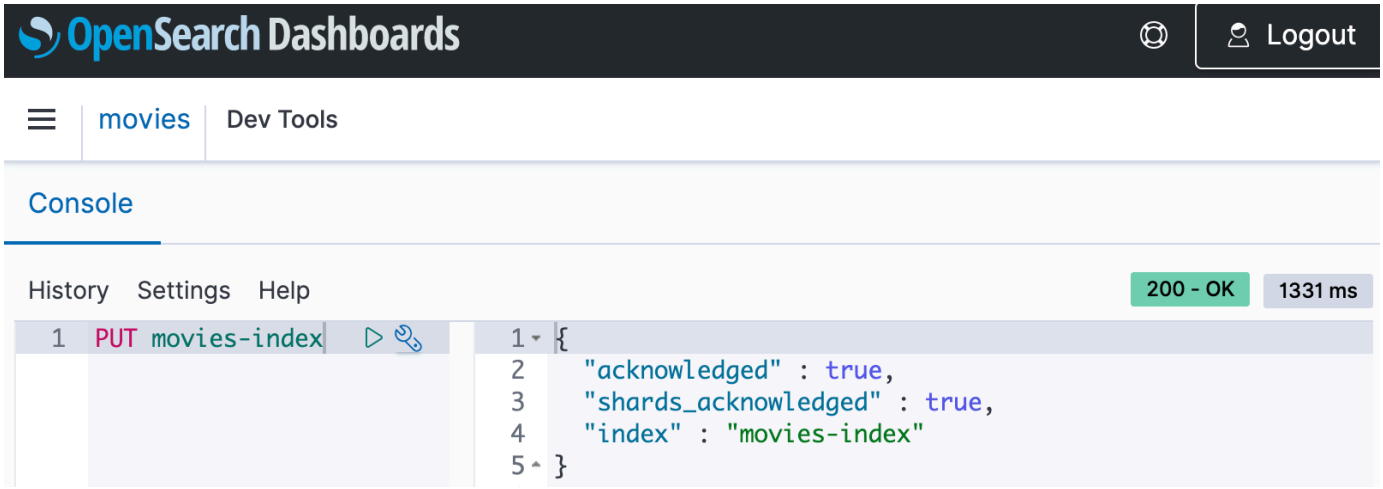
Paso 3: cargar y buscar datos

Puede cargar datos en una colección de OpenSearch sin servidor con Postman o cURL. Para ser breves, en estos ejemplos se utilizan las herramientas de desarrollo en la consola de OpenSearch Dashboards.

Para indexar y buscar datos en la colección de películas

1. Seleccione Colecciones en el panel de navegación de la izquierda y elija la colección de películas para abrir su página de detalles.
2. Seleccione la URL de OpenSearch Dashboards para la colección. La URL toma el formato `https://dashboards.{region}.aoss.amazonaws.com/_login/?collectionId={collection-id}`.
3. En OpenSearch Dashboards, abra el panel de navegación izquierdo y seleccione Herramientas de desarrollo.
4. Para crear un índice único llamado índice de películas, envíe la siguiente solicitud:

```
PUT movies-index
```



5. Para indexar un solo documento en índice de películas, envíe la siguiente solicitud:

```
PUT movies-index/_doc/1
{
  "title": "Shawshank Redemption",
  "genre": "Drama",
  "year": 1994
}
```

6. Para buscar datos en OpenSearch Dashboards, debe configurar al menos un patrón de índice. OpenSearch utiiza estos patrones para identificar los índices que desea analizar. Abra el menú principal del panel, seleccione Administración de pilas. Luego, seleccione Patrones de índices y, a continuación, seleccione Crear patrón de índice. Para este tutorial, escriba Películas.
7. seleccione Siguiente paso y, a continuación, seleccione Crear patrón de índice. Una vez creado el patrón, puede ver los diversos campos de documento, como `title` y `genre`.
8. Para comenzar a buscar los datos, abra el panel de navegación izquierdo y elija Explorar o utilice la API de búsqueda dentro de Herramientas de desarrollo.

Gestión de errores

Al ejecutar operaciones de indexación y búsqueda, usted puede recibir las siguientes respuestas de error:

- `HTTP 507`: indica que se produjo un error interno del servidor. Este error generalmente indica que sus unidades de cómputo de OpenSearch (OCU) están sobrecargadas por el volumen o la

complejidad de sus solicitudes. Aunque OpenSearch sin servidor escala automáticamente para administrar la carga, puede haber una demora en la implementación de recursos adicionales.

Para mitigar este error, implemente una política de reintentos con retroceso exponencial. Este enfoque reduce temporalmente la tasa de solicitudes para administrar la carga de manera eficaz. Para obtener más detalles, consulte Comportamiento de reintentos en la Guía de referencia de SDK y herramientas de AWS.

- `HTTP 402`: indica que usted alcanzó el límite máximo de la capacidad de unidades de cómputo de OpenSearch (OCU). Optimice su carga de trabajo para reducir el uso de OCU o solicite un aumento de cuota.

Paso 4: eliminar la colección

Dado que la colección de películas es para hacer pruebas, debe eliminarla cuando termine de experimentar.

Para eliminar una colección de OpenSearch sin servidor

1. Vuelva a la consola de Amazon OpenSearch Service.
2. Seleccione Colecciones en el panel de navegación de la izquierda y elija la colección de películas.
3. Seleccione Eliminar y confirme la eliminación.

Pasos a seguir a continuación

Ahora que sabe cómo crear una colección e indexar datos, puede que desee probar algunos de los siguientes ejercicios:

- Consulte opciones más avanzadas para crear una colección. Para obtener más información, consulte the section called “Administración de colecciones”.
- Aprenda a configurar las políticas de seguridad para gestionar la seguridad de las colecciones a escala. Para obtener más información, consulte the section called “Seguridad en OpenSearch sin servidor”.
- Descubra otras formas de indexar datos en las colecciones. Para obtener más información, consulte the section called “Ingesta de datos en las colecciones”.

Creación y administración de colecciones de Amazon OpenSearch sin servidor

Puede crear colecciones de Amazon OpenSearch sin servidor mediante la consola, la API y la AWS CLI, los AWS SDK y CloudFormation.

Temas

- Administración de colecciones de Amazon OpenSearch sin servidor
- Trabajo con colecciones de búsqueda vectorial
- Uso de políticas de ciclo de vida de los datos con Amazon OpenSearch sin servidor
- Uso de AWS SDK para interactuar con Amazon OpenSearch sin servidor
- Uso de CloudFormation para crear colecciones de Amazon OpenSearch sin servidor
- Hacer copias de seguridad de las colecciones mediante instantáneas

Administración de colecciones de Amazon OpenSearch sin servidor

Una colección de Amazon OpenSearch sin servidor es una agrupación lógica de uno o varios índices que representan una carga de trabajo de análisis. OpenSearch sin servidor administra y ajusta de forma automática la colección, lo que requiere una intervención manual mínima.

Temas

- Configuración de permisos para las colecciones
- Acerca del enriquecimiento semántico automático
- Creación de colecciones
- Acceso a OpenSearch Dashboards
- Visualización de colecciones
- Eliminación de colecciones

Configuración de permisos para las colecciones

OpenSearch sin servidor utiliza los siguientes permisos de AWS Identity and Access Management (IAM) para crear y administrar colecciones. Puede especificar las condiciones de IAM para restringir a los usuarios a colecciones específicas.

- `aoss:CreateCollection`: cree una colección.
- `aoss:ListCollections`: enumere las colecciones en la cuenta actual.
- `aoss:BatchGetCollection`: obtenga detalles sobre una o más colecciones.
- `aoss:UpdateCollection`: modifique una colección.
- `aoss:DeleteCollection`: elimine una colección.

El siguiente ejemplo de política de acceso basada en la identidad proporciona los permisos mínimos necesarios para que un usuario administre una única colección denominada `Logs`:

```
[
   {
      "Sid":"Allows managing logs collections",
      "Effect":"Allow",
      "Action":[
         "aoss:CreateCollection",
         "aoss:ListCollections",
         "aoss:BatchGetCollection",
         "aoss:UpdateCollection",
         "aoss:DeleteCollection",
         "aoss:CreateAccessPolicy",
         "aoss:CreateSecurityPolicy"
      ],
      "Resource":"*",
      "Condition":{
         "StringEquals":{
            "aoss:collection":"Logs"
         }
      }
   }
]
```

`aoss:CreateAccessPolicy` y `aoss:CreateSecurityPolicy` se incluyen porque se requieren las políticas de cifrado, de red y de acceso a los datos para que una colección funcione de forma correcta. Para más información, consulte the section called “Identity and Access Management”.

> **Note**
>
> Al crear la primera colección en su cuenta, también necesitará el permiso `iam:CreateServiceLinkedRole`. Para obtener más información, consulte the section called “Rol de creación de colecciones”.

Acerca del enriquecimiento semántico automático

Al crear o editar una colección, puede configurar el enriquecimiento semántico automático, lo que simplifica la implementación y las capacidades de la búsqueda semántica en Amazon OpenSearch Service. La búsqueda semántica arroja resultados de consulta que incorporan no solo la coincidencia de palabras clave, sino también la intención y el significado contextual de la búsqueda del usuario. Por ejemplo, si un usuario busca “cómo tratar un dolor de cabeza”, un sistema de búsqueda semántica puede arrojar los siguientes resultados:

- Remedios para la migraña
- Técnicas de manejo del dolor
- Analgésicos de venta libre
- Métodos naturales para aliviar el dolor de cabeza

El sistema entiende la intención subyacente incluso cuando estas frases exactas no estén en la consulta original.

El enriquecimiento semántico automático ofrece los siguientes beneficios:

Implementación simplificada

No necesita conocimiento experto en machine learning (ML) ni integraciones complejas.

Proceso automatizado

El enriquecimiento semántico se produce automáticamente durante la ingesta de datos.

Relevancia de búsqueda mejorada

El enriquecimiento semántico mejora la calidad y la precisión contextual de los resultados de búsqueda.

Escalabilidad

El enriquecimiento semántico aplica la búsqueda semántica a grandes conjuntos de datos sin intervención manual.

Funcionamiento

Para comenzar a utilizar el enriquecimiento semántico automático, cree o edite una colección y especifique qué campos de sus datos requieren capacidades de búsqueda semántica. Una vez que identifica los campos para la búsqueda semántica, a medida que los datos entran en OpenSearch Service, el proceso de enriquecimiento semántico automático enriquece automáticamente estos campos. Los datos enriquecidos permiten realizar búsquedas más inteligentes y sensibles al contexto.

> **Note**
>
> Tenga en cuenta los siguientes factores al implementar el enriquecimiento semántico automático:
>
> - Sobrecarga de procesamiento: el proceso de enriquecimiento puede prolongar el tiempo de procesamiento durante la ingesta.
> - Implicaciones de almacenamiento: los datos enriquecidos requieren espacio de almacenamiento adicional.
> - Limitaciones de idioma: compruebe si la opción multilingüe es compatible con los idiomas requeridos.

El enriquecimiento semántico automático para sistemas sin servidor ofrece las siguientes opciones de idioma.

Opción solo en inglés

- Optimizado para contenido en inglés
- Ideal para aplicaciones que se ocupan principalmente de texto en inglés

Opción multilingüe

- Admite los siguientes idiomas: árabe, bengalí, chino, inglés, finés, francés, hindi, indonesio, japonés, coreano, persa, ruso, español, swahili y telugu
- Perfecto para diversos contenidos internacionales o aplicaciones multilingües

Configuración de permisos para el enriquecimiento semántico automático

Antes de crear un índice de enriquecimiento semántico automatizado, debe configurar los permisos necesarios. En esta sección, se explican los permisos necesarios y cómo configurarlos.

Permisos de política de IAM

Utilice la siguiente política de AWS Identity and Access Management (IAM) para conceder los permisos necesarios a fin de trabajar con el enriquecimiento semántico automático:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AutomaticSemanticEnrichmentPermissions",
            "Effect": "Allow",
            "Action": [
                "aoss:CreateIndex",
                "aoss:GetIndex",
                "aoss:UpdateIndex",
                "aoss:DeleteIndex",
                "aoss:APIAccessAll"
            ],
            "Resource": "*"
        }
    ]
}
```

Permisos de claves

- Los permisos `aoss:*Index` permiten la administración de índices.
- El permiso `aoss:APIAccessAll` permite las operaciones de la API de OpenSearch.
- Para restringir los permisos a una colección específica, reemplace `"Resource": "*"` con el ARN de la colección.

Configurar los permisos de acceso a datos

A fin de configurar un índice para el enriquecimiento semántico automático, debe contar con las políticas de acceso a datos adecuadas que concedan permiso de acceso a los recursos de índices, canalizaciones y colecciones de modelos. Para obtener más información acerca de las políticas de acceso a datos, consulte Control de acceso a los datos para Amazon OpenSearch sin servidor. Para obtener información acerca del procedimiento de configuración de una política de acceso a datos, consulte Creación de políticas de acceso a los datos (consola).

Permisos de acceso a datos

```
[
    {
        "Description": "Create index permission",
        "Rules": [
            {
                "ResourceType": "index",
                "Resource": ["index/collection_name/*"],
                "Permission": [
                  "aoss:CreateIndex",
                  "aoss:DescribeIndex",
                  "aoss:UpdateIndex",
                  "aoss:DeleteIndex"
                ]
            }
        ],
        "Principal": [
            "arn:aws:iam::account_id:role/role_name"
        ]
    },
    {
        "Description": "Create pipeline permission",
        "Rules": [
            {
                "ResourceType": "collection",
                "Resource": ["collection/collection_name"],
                "Permission": [
                  "aoss:CreateCollectionItems",
                  "aoss:DescribeCollectionItems"
                ]
            }
        ],
        "Principal": [
```

```
            "arn:aws:iam::account_id:role/role_name"
        ]
    },
    {
        "Description": "Create model permission",
        "Rules": [
            {
                "ResourceType": "model",
                "Resource": ["model/collection_name/*"],
                "Permission": ["aoss:CreateMLResource"]
            }
        ],
        "Principal": [
            "arn:aws:iam::account_id:role/role_name"
        ]
    },
]
```

Permisos de acceso a la red

Para permitir que las API del servicio accedan a las colecciones privadas, debe configurar políticas de red que permitan el acceso necesario entre la API del servicio y la colección. Para obtener más información acerca de las políticas de red, consulte Acceso a la red para Amazon OpenSearch sin servidor.

```
[
   {
      "Description":"Enable automatic semantic enrichment in a private collection",
      "Rules":[
         {
            "ResourceType":"collection",
            "Resource":[
               "collection/collection_name"
            ]
         }
      ],
      "AllowFromPublic":false,
      "SourceServices":[
         "aoss.amazonaws.com"
      ],
   }
]
```

Configuración de los permisos de acceso a la red para una colección privada

1. Inicie sesión en la consola de OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. En el panel de navegación de la izquierda, seleccione Políticas de red. A continuación, realice alguna de las operaciones siguientes:
   - Elija un nombre de política existente y seleccione Editar.
   - Elija Crear política de red y configure los detalles de la política.
3. En el área Tipo de acceso, seleccione Privado (recomendado) y, a continuación, seleccione Acceso privado al servicio de AWS.
4. En el campo de búsqueda, seleccione Servicio y, a continuación, aoss.amazonaws.com.
5. En el área Tipo de recurso, seleccione la casilla Habilitar el acceso al punto de conexión de OpenSearch.
6. Para Buscar en colecciones o introducir términos prefijos específicos, en el campo de búsqueda, seleccione Nombre de la colección. A continuación, escriba o seleccione el nombre de las colecciones que desee asociar a la política de red.
7. Seleccione Crear para una nueva política de red o Actualizar para una política de red existente.

Creación de colecciones

Puede utilizar la consola o la AWS CLI para crear una colección sin servidor. En estos pasos se explica cómo crear una búsqueda o una colección de series temporales. Para crear una colección de búsquedas vectoriales, consulte the section called “Trabajo con colecciones de búsqueda vectorial”.

Temas

- Crear una colección (consola)
- Para crear una colección (CLI)

Crear una colección (consola)

Utilice los procedimientos de esta sección para crear una colección mediante la Consola de administración de AWS. En estos pasos se explica cómo crear una búsqueda o una colección de series temporales. Para crear una colección de búsquedas vectoriales, consulte the section called “Trabajo con colecciones de búsqueda vectorial”.

Temas

- Configurar los ajustes de la colección
- Configurar campos de búsqueda adicionales

Configurar los ajustes de la colección

Utilice el siguiente procedimiento para configurar la información de la colección.

Para configurar los ajustes de la colección mediante la consola

1. Vaya a la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home/.
2. Expanda Sin servidor en el panel de navegación de la izquierda y seleccione Colecciones.
3. Seleccione Crear colección.
4. Escriba un nombre y una descripción para la colección. El nombre debe cumplir los siguientes criterios:
   - Debe ser único para la cuenta y la Región de AWS
   - Contener solo letras minúsculas de la a a la z, números del 0 al 9 y un guion (-)
   - Contener entre 3 y 32 caracteres
5. Seleccione un tipo de colección:
   - Series temporales: segmento de análisis de registros que se centra en analizar grandes volúmenes de datos semiestructurados generados por máquinas. Se almacenan al menos 24 horas de datos en los índices calientes y el resto permanece en almacenamiento templado.
   - Búsqueda: búsqueda de texto completo que potencia las aplicaciones de sus redes internas y las aplicaciones orientadas a Internet. Todos los datos de búsqueda se almacenan en almacenamiento en caliente para garantizar un tiempo de respuesta rápido a las consultas.

     > **Note**
     >
     > Elija esta opción si va a habilitar la búsqueda semántica automática, tal y como se describe en Configurar los ajustes de la colección.

   - Búsqueda vectorial: búsqueda semántica en incrustaciones vectoriales que simplifica la administración de datos vectoriales. Impulsa las experiencias de búsqueda aumentada con

machine learning (ML) y las aplicaciones de IA generativa, como los chatbots, los asistentes personales y la detección de fraudes.

Para obtener más información, consulte the section called “Elección de un tipo de colección”.

6. En Tipo de implementación, elija la configuración de redundancia de la colección. De forma predeterminada, cada colección tiene redundancia, lo que significa que cada una de las unidades de computación de OpenSearch (OCU) de indexación y búsqueda tiene sus propias réplicas en espera en una zona de disponibilidad diferente. Para fines de desarrollo y pruebas, puede optar por deshabilitar la redundancia, lo que reduce a dos el número de OCU de su colección. Para obtener más información, consulte the section called “Funcionamiento”.
7. En Seguridad, elija Creación estándar.
8. En Cifrado, seleccione una clave de AWS KMS para cifrar los datos. OpenSearch sin servidor le notifica si el nombre de la colección que escribió coincide con un patrón definido en una política de cifrado. Puede optar por mantener esta coincidencia o anularla con una configuración de cifrado única. Para obtener más información, consulte the section called “Cifrado”.
9. En Configuración de acceso a la red, configure el acceso a la red para la colección.

   - En Tipo de acceso, seleccione acceso público o privado.

     Si elige el privado, especifique qué puntos de conexión de VPC y Servicios de AWS pueden acceder a la colección.

     - Puntos de conexión de VPC para el acceso: especifique uno o más puntos de conexión de VPC para permitir el acceso. Para crear un punto de conexión de VPC, consulte the section called “Puntos de conexión de VPC”.
     - Acceso privado a Servicio de AWS: seleccione uno o más servicios compatibles a los que permitir el acceso.
   - En Tipo de recurso, seleccione si los usuarios podrán acceder a la colección a través de su punto de conexión de OpenSearch (para realizar llamadas a la API a través de cURL y Postman, entre otros), a través del punto de conexión de OpenSearch Dashboards (para trabajar con las visualizaciones y realizar llamadas a la API a través de la consola) o mediante ambos.

> **Note**
>
> El acceso privado al Servicio de AWS se aplica solo al punto de conexión de OpenSearch, no al punto de conexión de OpenSearch Dashboards.

OpenSearch sin servidor le notifica si el nombre de la colección que escribió coincide con un patrón definido en una política de red. Puede optar por mantener esta coincidencia o anularla con una configuración de red personalizada. Para más información, consulte the section called "Acceso a la red".

10. (Opcional) Agregue una o varias etiquetas a la colección. Para más información, consulte the section called "Etiquetado de colecciones".
11. Elija Siguiente.

Configurar campos de búsqueda adicionales

Las opciones que aparecen en la página dos del flujo de trabajo de creación de colecciones dependen del tipo de colección que se esté creando. En esta sección, se describe cómo configurar campos de búsqueda adicionales para cada tipo de colección. En esta sección, también se describe cómo configurar el enriquecimiento semántico automático. Omita cualquier sección que no se aplique a su tipo de colección.

Temas

- Configurar el enriquecimiento semántico automático
- Configurar los campos de búsqueda de series temporales
- Configurar los campos de búsqueda léxica
- Configurar los campos de búsqueda vectorial

Configurar el enriquecimiento semántico automático

Al crear o editar una colección, puede configurar el enriquecimiento semántico automático, lo que simplifica la implementación y las capacidades de la búsqueda semántica en Amazon OpenSearch Service. La búsqueda semántica arroja resultados de consulta que incorporan no solo la coincidencia de palabras clave, sino también la intención y el significado contextual de la búsqueda del usuario. Para obtener más información, consulte Acerca del enriquecimiento semántico automático.

Para configurar el enriquecimiento semántico automático

1. En la sección Detalles del índice, en Nombre del índice, especifique un nombre.
2. En la sección Campos de enriquecimiento semántico automático, elija Agregar campo de búsqueda semántica.
3. En el campo Nombre del campo de entrada para el enriquecimiento semántico, escriba el nombre del campo que desea enriquecer.
4. El Tipo de datos es Texto. No puede cambiar este valor.
5. En Idioma, seleccione Inglés o Multilingüe.
6. Seleccione Agregar campo.
7. Cuando termine de configurar los campos opcionales para la colección, seleccione Siguiente. Revise los cambios y seleccione Enviar para crear la colección.

Configurar los campos de búsqueda de series temporales

Las opciones de la sección Campos de búsqueda de series temporales se refieren a los datos de series temporales y a los flujos de datos. Para obtener más información acerca de estos temas, consulte Administración de datos de series temporales en Amazon OpenSearch Service con Data Streams.

Para configurar los campos de búsqueda de series temporales

1. En la sección Campos de búsqueda de series temporales, seleccione Agregar campo de series temporales.
2. En Nombre del campo, escriba un nombre.
3. En Tipo de datos, seleccione un tipo de la lista.
4. Seleccione Agregar campo.
5. Cuando termine de configurar los campos opcionales para la colección, seleccione Siguiente. Revise los cambios y seleccione Enviar para crear la colección.

Configurar los campos de búsqueda léxica

La búsqueda léxica busca una coincidencia exacta entre una consulta de búsqueda y términos o palabras clave indexados.

Para configurar los campos de búsqueda léxica

1. En la sección Campos de búsqueda léxica, seleccione Agregar campo de búsqueda.
2. En Nombre del campo, escriba un nombre.
3. En Tipo de datos, seleccione un tipo de la lista.
4. Seleccione Agregar campo.
5. Cuando termine de configurar los campos opcionales para la colección, seleccione Siguiente. Revise los cambios y seleccione Enviar para crear la colección.

Configurar los campos de búsqueda vectorial

Para configurar los campos de búsqueda vectorial

1. En la sección Campos vectoriales, seleccione Agregar campo vectorial.
2. En Nombre del campo, escriba un nombre.
3. En Motor, seleccione un tipo de la lista.
4. Escriba el número de dimensiones.
5. En Métrica de distancia, seleccione un tipo de la lista.
6. Cuando termine de configurar los campos opcionales para la colección, seleccione Siguiente.
7. Revise los cambios y seleccione Enviar para crear la colección.

Para crear una colección (CLI)

Utilice los procedimientos de esta sección para crear una colección OpenSearch sin servidor mediante la AWS CLI.

Temas

- Antes de empezar
- Creación de una colección
- Crear una colección con un índice de enriquecimiento semántico automático

Antes de empezar

Antes de crear una colección mediante la AWS CLI, utilice el siguiente procedimiento a fin de crear las políticas necesarias para la colección.

> **Note**
>
> En cada uno de los siguientes procedimientos, al especificar un nombre para una colección, el nombre debe cumplir los siguientes criterios:
>
> - Debe ser único para la cuenta y la Región de AWS
> - Contener solo letras minúsculas de la a a la z, números del 0 al 9 y un guion (-)
> - Contener entre 3 y 32 caracteres

Para crear las políticas necesarias de una colección

1. Abra la AWS CLI y ejecute el siguiente comando para crear una política de cifrado con un patrón de recursos que coincida con el nombre previsto para la colección.

   ```
   aws opensearchserverless create-security-policy \
     --name policy name \
     --type encryption --policy "{\"Rules\":[{\"ResourceType\":\"collection\",
   \"Resource\":[\"collection\/collection name\"]}],\"AWSOwnedKey\":true}"
   ```

   Por ejemplo, si planea nombrar la colección como registros de la aplicación, puede crear una política de cifrado como esta:

   ```
   aws opensearchserverless create-security-policy \
     --name logs-policy \
     --type encryption --policy "{\"Rules\":[{\"ResourceType\":\"collection\",
   \"Resource\":[\"collection\/logs-application\"]}],\"AWSOwnedKey\":true}"
   ```

   Si piensa utilizar la política para colecciones adicionales, puede hacer que la regla sea más amplia, como `collection/logs*` o `collection/*`.

2. Ejecute el siguiente comando para configurar los ajustes de la red para la colección mediante una política de red. Es posible crear políticas de red después de crear una colección, pero le recomendamos que lo haga de antemano.

   ```
   aws opensearchserverless create-security-policy \
     --name policy name \
     --type network --policy "[{\"Description\":\"description\",\"Rules\":
   [{\"ResourceType\":\"dashboard\",\"Resource\":[\"collection\/collection name\"]},
   ```

```
{\"ResourceType\":\"collection\",\"Resource\":[\"collection\/collection name\"]}],
\"AllowFromPublic\":true}]"
```

Para seguir con el ejemplo anterior, logs-application, puede crear la siguiente política de red:

```
aws opensearchserverless create-security-policy \
  --name logs-policy \
  --type network --policy "[{\"Description\":\"Public access for logs collection
\",\"Rules\":[{\"ResourceType\":\"dashboard\",\"Resource\":[\"collection\/logs-
application\"]},{\"ResourceType\":\"collection\",\"Resource\":[\"collection\/logs-
application\"]}],\"AllowFromPublic\":true}]"
```

Creación de una colección

El siguiente procedimiento utiliza la acción de la API CreateCollection para crear una colección del tipo `SEARCH` o `TIMESERIES`. Si no especifica ningún tipo de colección en la solicitud, se usa `TIMESERIES` de forma predeterminada. Para obtener más información acerca de estos tipos, consulte Elección de un tipo de colección. Para crear una colección de búsquedas vectoriales, consulte the section called “Trabajo con colecciones de búsqueda vectorial”.

Si su colección está cifrada con un Clave propiedad de AWS, el `kmsKeyArn` es un `auto` en lugar de un ARN.

> **Important**
>
> Después de crear una colección, no podrá acceder a ella a menos que coincida con una política de acceso a los datos. Para obtener más información, consulte Control de acceso a los datos para Amazon OpenSearch sin servidor.

Creación de una colección

1. Compruebe que ha creado las políticas necesarias que se describen en Antes de empezar.
2. Ejecute el siguiente comando. En `type`, especifique `SEARCH` o `TIMESERIES`.

```
aws opensearchserverless create-collection --name "collection name" --
type collection type --description "description"
```

Crear una colección con un índice de enriquecimiento semántico automático

Utilice el siguiente procedimiento a fin de crear una nueva colección de OpenSearch sin servidor con un índice que esté configurado para el enriquecimiento semántico automático. El procedimiento utiliza la acción de la API CreateIndex de OpenSearch sin servidor.

Para crear una nueva colección con un índice configurado con el enriquecimiento semántico automático

Ejecute el siguiente comando para crear la colección y un índice.

```
aws opensearchserverless create-index \
--region Region ID \
--id collection name --index-name index name \
--index-schema \
'mapping in json'
```

A continuación se muestra un ejemplo.

```
aws opensearchserverless create-index \
--region us-east-1 \
--id conversation_history --index-name conversation_history_index \
--index-schema \
'{
    "mappings": {
        "properties": {
            "age": {
                "type": "integer"
            },
            "name": {
                "type": "keyword"
            },
            "user_description": {
                "type": "text"
            },
            "conversation_history": {
                "type": "text",
                "semantic_enrichment": {
                    "status": "ENABLED",
                    // Specifies the sparse tokenizer for processing multi-lingual text
                    "language_option": "MULTI-LINGUAL",
                    // If embedding_field is provided, the semantic embedding field
 will be set to the given name rather than original field name + "_embedding"
```

```
                    "embedding_field": "conversation_history_user_defined"
                }
            },
            "book_title": {
                "type": "text",
                "semantic_enrichment": {
                    // No embedding_field is provided, so the semantic embedding field
 is set to "book_title_embedding"
                    "status": "ENABLED",
                    "language_option": "ENGLISH"
                }
            },
            "abstract": {
                "type": "text",
                "semantic_enrichment": {
                    // If no language_option is provided, it will be set to English.
                    // No embedding_field is provided, so the semantic embedding field
 is set to "abstract_embedding"
                    "status": "ENABLED"
                }
            }
        }
    }
}'
```

Acceso a OpenSearch Dashboards

Después de crear una colección con la Consola de administración de AWS, puede ir hasta la URL de OpenSearch Dashboards de la colección. Para encontrar la URL de Dashboards, seleccione Colecciones en el panel de navegación izquierdo y elija la colección para abrir su página de detalles. La URL toma el formato `https://dashboards.us-east-1.aoss.amazonaws.com/_login/?collectionId=07tjusf2h91cunochc`. Cuando navegue hasta la URL, iniciará sesión automáticamente en Dashboards.

Si ya tiene disponible la URL de OpenSearch Dashboards, pero no está en la Consola de administración de AWS, si llama a la URL de los paneles desde el navegador, se redirigirá a la consola. Una vez que ingrese sus credenciales de AWS, iniciará sesión automáticamente en Dashboards. Para obtener información sobre cómo acceder a las colecciones para SAML, consulte Accessing OpenSearch Dashboards with SAML.

El tiempo de espera de la consola de OpenSearch Dashboards es de una hora y no se puede configurar.

> **Note**
>
> El 10 de mayo de 2023, OpenSearch introdujo un punto de conexión global común para OpenSearch Dashboards. Ahora puede navegar a OpenSearch Dashboards en el navegador con una URL que adopte el formato `https://dashboards.us-east-1.aoss.amazonaws.com/_login/?collectionId=07tjusf2h91cunochc`. Para garantizar la compatibilidad con versiones anteriores, seguiremos admitiendo los puntos de conexión de OpenSearch Dashboards específicos de la colección existente con el formato `https://07tjusf2h91cunochc.us-east-1.aoss.amazonaws.com/_dashboards`.

Visualización de colecciones

Puede ver las colecciones existentes en su Cuenta de AWS desde la pestaña Colecciones de la consola de Amazon OpenSearch Service.

Para enumerar las colecciones junto con sus ID, envíe la solicitud ListCollections.

```
aws opensearchserverless list-collections
```

Respuesta de ejemplo

```
{
   "collectionSummaries":[
      {
         "arn":"arn:aws:aoss:us-east-1:123456789012:collection/07tjusf2h91cunochc",
         "id":"07tjusf2h91cunochc",
         "name":"my-collection",
         "status":"CREATING"
      }
   ]
}
```

Para limitar los resultados de la búsqueda, utilice filtros de colección. Esta solicitud filtra la respuesta a las colecciones en estado `ACTIVE`:

```
aws opensearchserverless list-collections --collection-filters '{ "status": "ACTIVE" }'
```

Para obtener información detallada sobre una o más colecciones, incluidos el punto de conexión de OpenSearch y el punto de conexión de OpenSearch Dashboards, envíe la solicitud BatchGetCollection:

```
aws opensearchserverless batch-get-collection --ids ["07tjusf2h91cunochc",
 "1iu5usc4rame"]
```

> **Note**
>
> Puede incluir `--names` o `--ids` en la solicitud, pero no ambos.

Respuesta de ejemplo


```
{
   "collectionDetails":[
      {
         "id": "07tjusf2h91cunochc",
         "name": "my-collection",
         "status": "ACTIVE",
         "type": "SEARCH",
         "description": "",
         "arn": "arn:aws:aoss:us-east-1:123456789012:collection/07tjusf2h91cunochc",
         "kmsKeyArn": "arn:aws:kms:us-
east-1:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab",
         "createdDate": 1667446262828,
         "lastModifiedDate": 1667446300769,
         "collectionEndpoint": "https://07tjusf2h91cunochc.us-
east-1.aoss.amazonaws.com",
         "dashboardEndpoint": "https://07tjusf2h91cunochc.us-east-1.aoss.amazonaws.com/
_dashboards"
      },
      {
         "id": "178ukvtg3i82dvopdid",
         "name": "another-collection",
         "status": "ACTIVE",
         "type": "TIMESERIES",
         "description": "",
         "arn": "arn:aws:aoss:us-east-1:123456789012:collection/178ukvtg3i82dvopdid",
         "kmsKeyArn": "arn:aws:kms:us-
east-1:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab",
         "createdDate": 1667446262828,
```

```
         "lastModifiedDate": 1667446300769,
         "collectionEndpoint": "https://178ukvtg3i82dvopdid.us-
east-1.aoss.amazonaws.com",
         "dashboardEndpoint": "https://178ukvtg3i82dvopdid.us-
east-1.aoss.amazonaws.com/_dashboards"
      }
   ],
   "collectionErrorDetails":[]
}
```

Eliminación de colecciones

Al eliminar una colección, se eliminan todos los datos e índices en la colección. No se pueden recuperar las colecciones después de eliminarlas.

Para eliminar una colección mediante la consola

1. En el panel Colecciones de la consola de Amazon OpenSearch Service, seleccione la colección que desee eliminar.
2. Seleccione Eliminar y confirme la eliminación.

Para eliminar una colección mediante la AWS CLI, envíe la solicitud DeleteCollection:

```
aws opensearchserverless delete-collection --id 07tjusf2h91cunochc
```

Respuesta de ejemplo

```
{
   "deleteCollectionDetail":{
      "id":"07tjusf2h91cunochc",
      "name":"my-collection",
      "status":"DELETING"
   }
}
```

Trabajo con colecciones de búsqueda vectorial

El tipo de colección de búsqueda vectorial de OpenSearch sin servidor ofrece una capacidad de búsqueda por similitud que es escalable y de alto rendimiento. Le facilita la creación de experiencias

de búsqueda aumentada de machine learning (ML) modernas y aplicaciones de inteligencia artificial (IA) generativa sin tener que gestionar la infraestructura de base de datos vectorial subyacente.

Los casos de uso de las colecciones de búsquedas vectoriales incluyen búsquedas de imágenes, búsquedas de documentos, recuperación de música, recomendaciones de productos, búsquedas de vídeos, búsquedas basadas en la ubicación, detección de fraudes y detección de anomalías.

Como el motor vectorial de OpenSearch sin servidor utiliza la característica de búsqueda del vecino más cercano (k-NN) de OpenSearch, se obtiene la misma funcionalidad con la sencillez de un entorno sin servidor. El motor es compatible con la API del complemento k-NN. Con estas operaciones, puede sacar provecho de la búsqueda de texto completo, el filtrado avanzado, las agregaciones, las consultas geoespaciales, las consultas anidadas para una recuperación más rápida de los datos y la mejora de los resultados de búsqueda.

El motor vectorial proporciona métricas de distancia, como la distancia euclidiana, la similitud de coseno y la similitud de productos de puntos, y puede acomodar 16 000 dimensiones. Puede almacenar campos con varios tipos de datos para los metadatos, como números, valores booleanos, fechas, palabras clave y geopuntos. También puede almacenar campos con texto para obtener información descriptiva y añadir más contexto a los vectores almacenados. La ubicación de los tipos de datos reduce la complejidad, aumenta la capacidad de mantenimiento y evita la duplicación de datos, los problemas de compatibilidad de versiones y los problemas de licencia.

> **Note**
>
> Tenga en cuenta la siguiente información:
>
> - Amazon OpenSearch sin servidor admite la cuantificación escalar de 16 bits de Faiss, que se puede utilizar para realizar conversiones entre vectores flotantes de 32 bits y vectores de 16 bits. Para obtener más información, consulte Cuantificación escalar de 16 bits de Faiss. También puede utilizar vectores binarios para reducir los costos de memoria. Para obtener más información, consulte Vectores binarios.
> - Amazon OpenSearch sin servidor admite la búsqueda vectorial basada en disco. La búsqueda vectorial basada en disco reduce considerablemente los costos operativos de las cargas de trabajo de vector en entornos con poca memoria. Para obtener más información, consulte Búsqueda vectorial basada en disco.

Introducción a las colecciones de búsqueda vectorial

En este tutorial, debe completar los siguientes pasos para almacenar, buscar y recuperar incrustaciones vectoriales en tiempo real:

1. Configurar permisos
2. Crear una colección
3. Cargar y buscar datos
4. Eliminar la colección

Paso 1: configurar permisos

Para completar este tutorial (y para utilizar OpenSearch sin servidor en general), debe tener los permisos (de IAM) AWS Identity and Access Management correctos. En este tutorial, crea una colección, carga y busca datos y, a continuación, elimina la colección.

Su usuario o rol debe tener adjunta una política basada en la identidad con los siguientes permisos mínimos:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Action": [
        "aoss:CreateCollection",
        "aoss:ListCollections",
        "aoss:BatchGetCollection",
        "aoss:DeleteCollection",
        "aoss:CreateAccessPolicy",
        "aoss:ListAccessPolicies",
        "aoss:UpdateAccessPolicy",
        "aoss:CreateSecurityPolicy",
        "iam:ListUsers",
        "iam:ListRoles"
      ],
      "Effect": "Allow",
      "Resource": "*"
```

```
    }
  ]
}
```

Para obtener más información sobre los permisos de IAM de OpenSearch sin servidor, consulte the section called “Identity and Access Management”

Paso 2: crear una colección

Una colección es un grupo de índices de OpenSearch que funcionan juntos para soportar una carga de trabajo o un caso de uso específicos.

Para crear una colección de OpenSearch sin servidor

1. Abra la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. Seleccione Colecciones en el panel de navegación de la izquierda y elija Crear colección.
3. Asigne un nombre a la carcasa de la colección.
4. Para el tipo de colección, seleccione Búsqueda vectorial. Para obtener más información, consulte the section called “Elección de un tipo de colección”.
5. En Tipo de implementación, elimine Habilitar redundancia (réplicas activas). Esto crea una colección en modo de desarrollo o prueba y reduce la cantidad de unidades de computación de OpenSearch (OCU) de la colección a dos. Si desea crear un entorno de producción en este tutorial, deje la casilla seleccionada.
6. En Seguridad, seleccione Crear fácilmente para optimizar la configuración de seguridad. De forma predeterminada, todos los datos del motor vectorial se cifran en tránsito y en reposo. El motor vectorial admite permisos de IAM detallados para que pueda definir quién puede crear, actualizar y eliminar cifrados, redes, colecciones e índices.
7. Seleccione Siguiente.
8. Revise la configuración de la colección y seleccione Enviar. Espere unos minutos hasta que el estado de la colección cambie a `Active`.

Paso 3: cargar y buscar datos

Un índice es una colección de documentos con un esquema de datos común que permite almacenar, buscar y recuperar las incrustaciones vectoriales y otros campos. Puede crear y cargar datos en los índices de una colección de OpenSearch sin servidor mediante la consola de Herramientas para

desarrolladores en OpenSearch Dashboards, o una herramienta HTTP como Postman o awscurl. En este tutorial se utilizan Herramientas para desarrolladores.

Para indexar y buscar datos en la colección de películas

1. Para crear un índice único para su nueva colección, envíe la siguiente solicitud en la consola de Herramientas para desarrolladores. De forma predeterminada, se crea un índice con un motor `nmslib` y una distancia euclidiana.

```
PUT housing-index
{
   "settings": {
      "index.knn": true
   },
   "mappings": {
      "properties": {
         "housing-vector": {
            "type": "knn_vector",
            "dimension": 3
         },
         "title": {
            "type": "text"
         },
         "price": {
            "type": "long"
         },
         "location": {
            "type": "geo_point"
         }
      }
   }
}
```

2. Para indexar un solo documento en housing-index, envíe la siguiente solicitud:

```
POST housing-index/_doc
{
  "housing-vector": [
    10,
    20,
    30
  ],
  "title": "2 bedroom in downtown Seattle",
```

```
  "price": "2800",
  "location": "47.71, 122.00"
}
```

3. Para buscar propiedades similares a las del índice, envíe la siguiente consulta:

```
GET housing-index/_search
{
    "size": 5,
    "query": {
        "knn": {
            "housing-vector": {
                "vector": [
                    10,
                    20,
                    30
                ],
                "k": 5
            }
        }
    }
}
```

Paso 4: eliminar la colección

Dado que la colección de housing es para hacer pruebas, debe eliminarla cuando termine de experimentar.

Para eliminar una colección de OpenSearch sin servidor

1. Vuelva a la consola de Amazon OpenSearch Service.
2. Seleccione Colecciones en el panel de navegación izquierdo y elija la colección de propiedades.
3. Para confirmar la eliminación, seleccione Eliminar.

Búsqueda filtrada

Puede usar filtros para mejorar los resultados de búsqueda semántica. Para crear un índice y realizar una búsqueda filtrada en sus documentos, sustituya los datos de carga y búsqueda en el tutorial anterior por las siguientes instrucciones. Los demás pasos siguen siendo los mismos. Para obtener más información acerca de los filtros, consulte búsqueda k-NN con filtros.

Para indexar y buscar datos en la colección de películas

1. Para crear un índice único para su colección, envíe la siguiente solicitud en la consola de Herramientas para desarrolladores:

```
PUT housing-index-filtered
{
  "settings": {
    "index.knn": true
  },
  "mappings": {
    "properties": {
      "housing-vector": {
        "type": "knn_vector",
        "dimension": 3,
        "method": {
          "engine": "faiss",
          "name": "hnsw"
        }
      },
      "title": {
        "type": "text"
      },
      "price": {
        "type": "long"
      },
      "location": {
        "type": "geo_point"
      }
    }
  }
}
```

2. Para indexar un solo documento en housing-index-filtered, envíe la siguiente solicitud:

```
POST housing-index-filtered/_doc
{
  "housing-vector": [
    10,
    20,
    30
  ],
  "title": "2 bedroom in downtown Seattle",
```

```
  "price": "2800",
  "location": "47.71, 122.00"
}
```

3. Para buscar los datos de un apartamento en Seattle por un precio determinado y dentro de una distancia determinada de un punto geográfico, envíe la siguiente solicitud:

```
GET housing-index-filtered/_search
{
  "size": 5,
  "query": {
    "knn": {
      "housing-vector": {
        "vector": [
          0.1,
          0.2,
          0.3
        ],
        "k": 5,
        "filter": {
          "bool": {
            "must": [
              {
                "query_string": {
                  "query": "Find me 2 bedroom apartment in Seattle under $3000 ",
                  "fields": [
                    "title"
                  ]
                }
              },
              {
                "range": {
                  "price": {
                    "lte": 3000
                  }
                }
              },
              {
                "geo_distance": {
                  "distance": "100miles",
                  "location": {
                    "lat": 48,
                    "lon": 121
```

```
                    }
                  }
                }
              ]
            }
          }
        }
      }
    }
  }
}
```

Cargas de trabajo a escala de mil millones

Las colecciones de búsquedas vectoriales admiten cargas de trabajo con miles de millones de vectores. No necesita reindexar para escalar porque el escalado automático lo hace por usted. Si tiene millones de vectores (o más) con un gran número de dimensiones y necesita más de 200 OCU, póngase en contacto con AWS Support para aumentar el máximo de unidades de computación de OpenSearch (OCU) para su cuenta.

Limitaciones

Las colecciones de búsqueda vectorial presentan las siguientes limitaciones:

- Las colecciones de búsqueda vectorial no son compatibles con el motor ANN Apache Lucene.
- Las colecciones de búsqueda vectorial solo admiten el algoritmo HNSW con Faiss y no admiten FIV ni IVFQ.
- Las colecciones de búsqueda vectorial no admiten las operaciones de la API de calentamiento, estadística y entrenamiento de modelos.
- Las colecciones de búsqueda vectorial no admiten scripts en línea ni almacenados.
- La información sobre el recuento de índices no está disponible en las Consola de administración de AWS de búsquedas vectoriales.
- El intervalo de actualización de los índices de las colecciones de búsqueda vectorial es de 60 segundos.

Pasos a seguir a continuación

Ahora que sabe cómo crear una colección de búsquedas vectoriales e indexar datos, puede que desee probar algunos de los siguientes ejercicios:

- Utilice el cliente OpenSearch Python para trabajar con colecciones de búsqueda vectorial. Consulte este tutorial en GitHub.
- Utilice el cliente OpenSearch Java para trabajar con colecciones de búsqueda vectorial. Consulte este tutorial en GitHub.
- Configure LangChain para usar OpenSearch como almacén de vectores. LangChain es un marco de código abierto para desarrollar aplicaciones basadas en modelos de lenguaje. Para obtener más información, consulte Documentación de LangChain.

Uso de políticas de ciclo de vida de los datos con Amazon OpenSearch sin servidor

Una política del ciclo de vida de los datos en Amazon OpenSearch sin servidor define durante cuánto tiempo OpenSearch sin servidor retiene los datos de una colección de series temporales. Por ejemplo, puede establecer una política para retener los datos de registro durante 30 días antes de que OpenSearch sin servidor los elimine.

Puede configurar una política independiente para cada índice dentro de cada colección de series temporales de su Cuenta de AWS. OpenSearch sin servidor retiene los documentos durante al menos el tiempo que se especifique en la política. A continuación, borra automáticamente los documentos según lo mejor que pueda, normalmente en un plazo de 48 horas o en un 10 % del periodo de retención, lo que sea más largo.

Solo las recopilaciones de series temporales admiten las políticas del ciclo de vida de los datos. Las colecciones de búsqueda y búsqueda vectorial no las admiten.

Temas

- Políticas de ciclo de vida de los datos
- Permisos necesarios
- Prioridad política
- Sintaxis de la política
- Creación de políticas del ciclo de vida de los datos
- Actualización de políticas de ciclo de vida de los datos
- Eliminación de políticas de ciclo de vida de los datos

Políticas de ciclo de vida de los datos

En una política de ciclo de vida de los datos, especifique una serie de reglas. La política de ciclo de vida de los datos le permite administrar el período de retención de los datos asociados a los índices o colecciones que cumplen estas reglas. Estas reglas definen el período de retención de los datos de un índice o grupo de índices. Cada regla consta de un tipo de recurso (`index`), un período de retención y una lista de recursos (índices) a los que se aplica el período de retención.

El período de retención se define con uno de los siguientes formatos:

- `"MinIndexRetention": "24h"`: OpenSearch sin servidor conserva los datos del índice durante el período especificado en horas o días. Puede configurar este período para que sea de `24h` a `3650d`.
- `"NoMinIndexRetention": true`: OpenSearch sin servidor conserva los datos del índice de forma indefinida.

En el siguiente ejemplo de política, la primera regla especifica un período de retención de 15 días para todos los índices de la colección `marketing`. La segunda regla especifica que todos los nombres de índice que comiencen por `log` en la colección `finance` no tienen un período de retención establecido y se conservarán indefinidamente.

```
{
   "lifeCyclePolicyDetail": {
      "type": "retention",
      "name": "my-policy",
      "policyVersion": "MTY4ODI0NTM2OTk1N18x",
      "policy": {
         "Rules": [
            {
            "ResourceType":"index",
            "Resource":[
               "index/marketing/*"
            ],
            "MinIndexRetention": "15d"
         },
         {
            "ResourceType":"index",
            "Resource":[
               "index/finance/log*"
            ],
```

```
            "NoMinIndexRetention": true
         }
         ]
      },
      "createdDate": 1688245369957,
      "lastModifiedDate": 1688245369957
   }
}
```

En el siguiente ejemplo de regla de política, OpenSearch sin servidor conserva indefinidamente los datos de todos los índices de todas las colecciones de la cuenta.

```
{
   "Rules": [
      {
         "ResourceType": "index",
         "Resource": [
            "index/*/*"
         ]
      }
   ],
   "NoMinIndexRetention": true
}
```

Permisos necesarios

Las políticas de ciclo de vida para OpenSearch sin servidor utilizan los siguientes permisos AWS Identity and Access Management (IAM). Puede especificar condiciones de IAM para restringir a los usuarios a las políticas de ciclo de vida de los datos asociadas con colecciones e índices específicos.

- `aoss:CreateLifecyclePolicy`: cree una política de ciclo de vida de los datos.
- `aoss:ListLifecyclePolicies`: enumere todas las políticas de ciclo de vida de los datos de la cuenta actual.
- `aoss:BatchGetLifecyclePolicy`: consulte una política del ciclo de vida de los datos asociada a un nombre de cuenta o política.
- `aoss:BatchGetEffectiveLifecyclePolicy`: consulte una política del ciclo de vida de los datos para un recurso determinado (`index` es el único recurso compatible).
- `aoss:UpdateLifecyclePolicy`: modifique una política de ciclo de vida de los datos determinada y cambie su configuración o recurso de retención.

- `aoss:DeleteLifecyclePolicy`: elimine una política de ciclo de vida de los datos.

La siguiente política de acceso basada en identidades permite al usuario ver todas las políticas de ciclo de vida de los datos y actualizarlas según el patrón de recursos `index/application-logs`:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "aoss:UpdateLifecyclePolicy"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aoss:collection": "application-logs"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "aoss:ListLifecyclePolicies",
                "aoss:BatchGetLifecyclePolicy"
            ],
            "Resource": "*"
        }
    ]
}
```

Prioridad política

Puede haber situaciones en las que las reglas de la política de ciclo de vida de los datos se superpongan, dentro de las políticas o entre ellas. Cuando esto sucede, una regla con un nombre o patrón de recurso más específico para un índice anula una regla con un nombre o patrón de recurso más general para cualquier índice que sea común a ambas reglas.

Por ejemplo, en la siguiente política, se aplican dos reglas a un índice `index/sales/logstash`. En esta situación, la segunda regla tiene prioridad porque `index/sales/log*` es la que coincide más tiempo con `index/sales/logstash`. Por lo tanto, OpenSearch sin servidor no establece ningún período de retención para el índice.

```
{
      "Rules":[
         {
            "ResourceType":"index",
            "Resource":[
               "index/sales/*",
            ],
            "MinIndexRetention": "15d"
         },
         {
            "ResourceType":"index",
            "Resource":[
               "index/sales/log*",
            ],
            "NoMinIndexRetention": true
         }
      ]
   }
```

Sintaxis de la política

Proporcione una o más reglas. Estas reglas definen la configuración del ciclo de vida de los datos para los índices de OpenSearch sin servidor.

Cada regla contiene los siguientes elementos. Puede proporcionar `MinIndexRetention` o `NoMinIndexRetention` en cada regla, pero no ambas.

Elemento	Descripción
Tipo de recurso	El tipo de recurso al que se aplica la regla. La única opción admitida para las políticas del ciclo de vida de los datos es `index`.
Recurso	Una lista de nombres o patrones de recursos. Los patrones son un prefijo seguido de un comodín (*), que permiten que los permisos

Elemento	Descripción
	asociados se apliquen a varios recursos. Por ejemplo, `index/<collection-name|pattern> /<index-name|pattern>` .
Retención mínima de índices	El período mínimo, en días (d) u horas (h), para retener el documento en el índice. El límite inferior es `24h` y el límite superior es `3650d`.
Sin retención mínima de índices	Si `true`, OpenSearch sin servidor conserva los documentos indefinidamente.

En el siguiente ejemplo, la primera regla se aplica a todos los índices incluidos en el patrón `autoparts-inventory` (`index/autoparts-inventory/*`) y exige que los datos se conserven durante al menos 20 días antes de que se pueda realizar cualquier acción, como eliminarlos o archivarlos.

La segunda regla se dirige a los índices que coinciden con el patrón `auto*/gear` (`index/auto*/gear`) y establece un periodo de retención mínimo de 24 horas.

La tercera regla se aplica específicamente al índice `tires` y no tiene un periodo mínimo de retención, lo que significa que los datos de este índice se pueden eliminar o archivar de forma inmediata o en función de otros criterios. Estas reglas ayudan a administrar la retención de los datos del índice con tiempos de retención variables o sin restricciones de retención.

```
{
  "Rules": [
    {
      "ResourceType": "index",
      "Resource": [
        "index/autoparts-inventory/*"
      ],
      "MinIndexRetention": "20d"
    },
    {
      "ResourceType": "index",
      "Resource": [
        "index/auto*/gear"
      ],
```

```
      "MinIndexRetention": "24h"
    },
    {
      "ResourceType": "index",
      "Resource": [
        "index/autoparts-inventory/tires"
      ],
      "NoMinIndexRetention": true
    }
  ]
}
```

Creación de políticas del ciclo de vida de los datos

Para crear una política del ciclo de vida de los datos, debe definir reglas que administren la retención y la eliminación de los datos en función de criterios específicos.

Consola

Para crear una política del ciclo de vida de los datos

1. Inicie sesión en la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. En el panel de navegación de la izquierda, seleccione Políticas de ciclo de vida de los datos.
3. Seleccione Creación de una política de ciclo de vida de los datos.
4. Escriba un nombre descriptivo para la política.
5. Para Ciclo de vida de los datos, elija Agregar y seleccione las colecciones e índices de la política.

   Comience por elegir las colecciones a las que pertenecen los índices. A continuación, seleccione el índice de la lista o ingrese un patrón de índice. Para seleccionar todas las colecciones como orígenes, escriba un asterisco (*).
6. Para Retención de datos, puede optar por retener los datos indefinidamente o desmarcar Ilimitada (nunca eliminar) y especificar un periodo de tiempo después del cual OpenSearch sin servidor eliminará automáticamente los datos de Amazon S3.
7. Seleccione Guardar y, a continuación, Crear.

AWS CLI

Para crear una política del ciclo de vida de los datos mediante la AWS CLI, utilice el comando create-lifecycle-policy con las siguientes opciones:

- `--name`: el nombre de la política.
- `--type`: el tipo de política. El único valor disponible actualmente es `retention`.
- `--policy`: la política del ciclo de vida de los datos. Este parámetro acepta tanto políticas insertadas como archivos .json. Debe codificar las políticas insertadas como una cadena de escape de JSON. Para proporcionar la política en un archivo, utilice el formato `--policy file://my-policy.json`.

Example

```
aws opensearchserverless create-lifecycle-policy \
  --name my-policy \
  --type retention \
  --policy "{\"Rules\":[{\"ResourceType\":\"index\",\"Resource\":[\"index/autoparts-inventory/*\"],\"MinIndexRetention\": \"81d\"},{\"ResourceType\":\"index\",\"Resource\":[\"index/sales/orders*\"],\"NoMinIndexRetention\":true}]}"
```

Actualización de políticas de ciclo de vida de los datos

Para actualizar una política del ciclo de vida de los datos, puede modificar las reglas existentes para reflejar los cambios en sus requisitos de retención o eliminación de datos. Esto le permite adaptar sus políticas a medida que evolucionen sus necesidades de administración de datos.

Pueden transcurrir unos minutos de retraso entre el momento en que se actualiza la política y el momento en que OpenSearch sin servidor comienza a aplicar los nuevos periodos de retención.

Consola

Para actualizar una política del ciclo de vida de los datos

1. Inicie sesión en la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. En el panel de navegación de la izquierda, seleccione Políticas de ciclo de vida de los datos.
3. Seleccione la política del ciclo de vida de los datos que desee actualizar y, a continuación, elija Editar.

4. Modifique la política mediante el editor visual o el editor JSON.
5. Seleccione Save.

AWS CLI

Para actualizar una política del ciclo de vida de los datos mediante la AWS CLI, utilice el comando update-lifecycle-policy.

Debe incluir el parámetro `--policy-version` en la solicitud. Puede recuperar la versión de la política mediante los comandos list-lifecycle-policies o batch-get-lifecycle-policy. Recomendamos incluir la versión más reciente de la política para evitar que se sobrescriban accidentalmente los cambios de otros usuarios.

La siguiente solicitud actualiza una política del ciclo de vida de los datos con un nuevo documento JSON de política.

Example

```
aws opensearchserverless update-lifecycle-policy \
  --name my-policy \
  --type retention \
  --policy-version MTY2MzY5MTY1MDA3Ml8x \
  --policy file://my-new-policy.json
```

Eliminación de políticas de ciclo de vida de los datos

Al eliminar una política del ciclo de vida de los datos, OpenSearch sin servidor deja de aplicarla a los índices coincidentes.

Consola

Para eliminar una política del ciclo de vida de los datos

1. Inicie sesión en la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. En el panel de navegación de la izquierda, seleccione Políticas de ciclo de vida de los datos.
3. Seleccione la política de configuración que desea eliminar y, a continuación, elija Eliminar y confirme la eliminación.

AWS CLI

Para eliminar una política del ciclo de vida de los datos mediante la AWS CLI, utilice el comando delete-lifecycle-policy.

Example

```
aws opensearchserverless delete-lifecycle-policy \
  --name my-policy \
  --type retention
```

Uso de AWS SDK para interactuar con Amazon OpenSearch sin servidor

En esta sección, se incluyen ejemplos de cómo utilizar los AWS SDK para interactuar con la API de configuración de Amazon OpenSearch sin servidor. Estos ejemplos de código muestran cómo crear políticas de seguridad y colecciones, y cómo consultar colecciones.

> **Note**
>
> Actualmente estamos creando estos ejemplos de código. Si quieres contribuir con un ejemplo de código (Java, Go, etc.), abra una solicitud de extracción directamente en el Repositorio de GitHub.

Temas

- Python
- JavaScript

Python

El siguiente script de ejemplo utiliza AWS SDK para Python (Boto3), así como también el cliente para Python opensearch-py para crear políticas de acceso a datos, de cifrado y de redes, crear colecciones de coincidencias e indexar algunos datos de ejemplo.

Ejecute los siguientes comandos para asegurarse de que dispone de todas las dependencias necesarias:

```
pip install opensearch-py
```

```
pip install boto3
pip install botocore
pip install requests-aws4auth
```

En el script, reemplace `Principal` por el nombre de recurso de Amazon (ARN) del usuario o rol que firma la solicitud. También puede modificar opcionalmente el archivo `region`.

```
from opensearchpy import OpenSearch, RequestsHttpConnection
from requests_aws4auth import AWS4Auth
import boto3
import botocore
import time

# Build the client using the default credential configuration.
# You can use the CLI and run 'aws configure' to set access key, secret
# key, and default region.

client = boto3.client('opensearchserverless')
service = 'aoss'
region = 'us-east-1'
credentials = boto3.Session().get_credentials()
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key,
                   region, service, session_token=credentials.token)


def createEncryptionPolicy(client):
    """Creates an encryption policy that matches all collections beginning with tv-"""
    try:
        response = client.create_security_policy(
            description='Encryption policy for TV collections',
            name='tv-policy',
            policy="""
                {
                    \"Rules\":[
                        {
                            \"ResourceType\":\"collection\",
                            \"Resource\":[
                                \"collection\/tv-*\"
                            ]
                        }
                    ],
                    \"AWSOwnedKey\":true
                }
```

```
                """,
            type='encryption'
        )
        print('\nEncryption policy created:')
        print(response)
    except botocore.exceptions.ClientError as error:
        if error.response['Error']['Code'] == 'ConflictException':
            print(
                '[ConflictException] The policy name or rules conflict with an existing
 policy.')
        else:
            raise error


def createNetworkPolicy(client):
    """Creates a network policy that matches all collections beginning with tv-"""
    try:
        response = client.create_security_policy(
            description='Network policy for TV collections',
            name='tv-policy',
            policy="""
                [{
                    \"Description\":\"Public access for TV collection\",
                    \"Rules\":[
                        {
                            \"ResourceType\":\"dashboard\",
                            \"Resource\":[\"collection\/tv-*\"]
                        },
                        {
                            \"ResourceType\":\"collection\",
                            \"Resource\":[\"collection\/tv-*\"]
                        }
                    ],
                    \"AllowFromPublic\":true
                }]
                """,
            type='network'
        )
        print('\nNetwork policy created:')
        print(response)
    except botocore.exceptions.ClientError as error:
        if error.response['Error']['Code'] == 'ConflictException':
            print(
                '[ConflictException] A network policy with this name already exists.')
```

```
        else:
            raise error


def createAccessPolicy(client):
    """Creates a data access policy that matches all collections beginning with tv-"""
    try:
        response = client.create_access_policy(
            description='Data access policy for TV collections',
            name='tv-policy',
            policy="""
                [{
                    \"Rules\":[
                        {
                            \"Resource\":[
                                \"index\/tv-*\/*\"
                            ],
                            \"Permission\":[
                                \"aoss:CreateIndex\",
                                \"aoss:DeleteIndex\",
                                \"aoss:UpdateIndex\",
                                \"aoss:DescribeIndex\",
                                \"aoss:ReadDocument\",
                                \"aoss:WriteDocument\"
                            ],
                            \"ResourceType\": \"index\"
                        },
                        {
                            \"Resource\":[
                                \"collection\/tv-*\"
                            ],
                            \"Permission\":[
                                \"aoss:CreateCollectionItems\"
                            ],
                            \"ResourceType\": \"collection\"
                        }
                    ],
                    \"Principal\":[
                        \"arn:aws:iam::123456789012:role\/Admin\"
                    ]
                }]
                """,
            type='data'
        )
```

```
        print('\nAccess policy created:')
        print(response)
    except botocore.exceptions.ClientError as error:
        if error.response['Error']['Code'] == 'ConflictException':
            print(
                '[ConflictException] An access policy with this name already exists.')
        else:
            raise error


def createCollection(client):
    """Creates a collection"""
    try:
        response = client.create_collection(
            name='tv-sitcoms',
            type='SEARCH'
        )
        return(response)
    except botocore.exceptions.ClientError as error:
        if error.response['Error']['Code'] == 'ConflictException':
            print(
                '[ConflictException] A collection with this name already exists. Try
 another name.')
        else:
            raise error


def waitForCollectionCreation(client):
    """Waits for the collection to become active"""
    response = client.batch_get_collection(
        names=['tv-sitcoms'])
    # Periodically check collection status
    while (response['collectionDetails'][0]['status']) == 'CREATING':
        print('Creating collection...')
        time.sleep(30)
        response = client.batch_get_collection(
            names=['tv-sitcoms'])
    print('\nCollection successfully created:')
    print(response["collectionDetails"])
    # Extract the collection endpoint from the response
    host = (response['collectionDetails'][0]['collectionEndpoint'])
    final_host = host.replace("https://", "")
    indexData(final_host)
```

```
def indexData(host):
    """Create an index and add some sample data"""
    # Build the OpenSearch client
    client = OpenSearch(
        hosts=[{'host': host, 'port': 443}],
        http_auth=awsauth,
        use_ssl=True,
        verify_certs=True,
        connection_class=RequestsHttpConnection,
        timeout=300
    )
    # It can take up to a minute for data access rules to be enforced
    time.sleep(45)

    # Create index
    response = client.indices.create('sitcoms-eighties')
    print('\nCreating index:')
    print(response)

    # Add a document to the index.
    response = client.index(
        index='sitcoms-eighties',
        body={
            'title': 'Seinfeld',
            'creator': 'Larry David',
            'year': 1989
        },
        id='1',
    )
    print('\nDocument added:')
    print(response)


def main():
    createEncryptionPolicy(client)
    createNetworkPolicy(client)
    createAccessPolicy(client)
    createCollection(client)
    waitForCollectionCreation(client)


if __name__ == "__main__":
```

```
    main()
```

JavaScript

El siguiente script de ejemplo utiliza el SDK para JavaScript en Node.js, así como también el cliente para JavaScript opensearch-js para crear políticas de acceso a datos, de cifrado y de redes, crear colecciones de coincidencias, crear un índice e indexar algunos datos de ejemplo.

Ejecute los siguientes comandos para asegurarse de que dispone de todas las dependencias necesarias:

```
npm i aws-sdk
npm i aws4
npm i @opensearch-project/opensearch
```

En el script, reemplace `Principal` por el nombre de recurso de Amazon (ARN) del usuario o rol que firma la solicitud. También puede modificar opcionalmente el archivo `region`.

```
var AWS = require('aws-sdk');
var aws4 = require('aws4');
var {
    Client,
    Connection
} = require("@opensearch-project/opensearch");
var {
    OpenSearchServerlessClient,
    CreateSecurityPolicyCommand,
    CreateAccessPolicyCommand,
    CreateCollectionCommand,
    BatchGetCollectionCommand
} = require("@aws-sdk/client-opensearchserverless");
var client = new OpenSearchServerlessClient();

async function execute() {
    await createEncryptionPolicy(client)
    await createNetworkPolicy(client)
    await createAccessPolicy(client)
    await createCollection(client)
    await waitForCollectionCreation(client)
}

async function createEncryptionPolicy(client) {
```

```
    // Creates an encryption policy that matches all collections beginning with 'tv-'
    try {
        var command = new CreateSecurityPolicyCommand({
            description: 'Encryption policy for TV collections',
            name: 'tv-policy',
            type: 'encryption',
            policy: " \
        { \
            \"Rules\":[ \
                { \
                    \"ResourceType\":\"collection\", \
                    \"Resource\":[ \
                        \"collection\/tv-*\" \
                    ] \
                } \
            ], \
            \"AWSOwnedKey\":true \
        }"
        });
        const response = await client.send(command);
        console.log("Encryption policy created:");
        console.log(response['securityPolicyDetail']);
    } catch (error) {
        if (error.name === 'ConflictException') {
            console.log('[ConflictException] The policy name or rules conflict with an
 existing policy.');
        } else
            console.error(error);
    };
}

async function createNetworkPolicy(client) {
    // Creates a network policy that matches all collections beginning with 'tv-'
    try {
        var command = new CreateSecurityPolicyCommand({
            description: 'Network policy for TV collections',
            name: 'tv-policy',
            type: 'network',
            policy: " \
            [{ \
                \"Description\":\"Public access for television collection\", \
                \"Rules\":[ \
                    { \
                        \"ResourceType\":\"dashboard\", \
```

```
                        \"Resource\":[\"collection\/tv-*\"] \
                    }, \
                    { \
                        \"ResourceType\":\"collection\", \
                        \"Resource\":[\"collection\/tv-*\"] \
                    } \
                ], \
                \"AllowFromPublic\":true \
            }]"
        });
        const response = await client.send(command);
        console.log("Network policy created:");
        console.log(response['securityPolicyDetail']);
    } catch (error) {
        if (error.name === 'ConflictException') {
            console.log('[ConflictException] A network policy with that name already
 exists.');
        } else
            console.error(error);
    };
}

async function createAccessPolicy(client) {
    // Creates a data access policy that matches all collections beginning with 'tv-'
    try {
        var command = new CreateAccessPolicyCommand({
            description: 'Data access policy for TV collections',
            name: 'tv-policy',
            type: 'data',
            policy: " \
            [{ \
                \"Rules\":[ \
                    { \
                        \"Resource\":[ \
                            \"index\/tv-*\/*\" \
                        ], \
                        \"Permission\":[ \
                            \"aoss:CreateIndex\", \
                            \"aoss:DeleteIndex\", \
                            \"aoss:UpdateIndex\", \
                            \"aoss:DescribeIndex\", \
                            \"aoss:ReadDocument\", \
                            \"aoss:WriteDocument\" \
                        ], \
```

```
                        \"ResourceType\": \"index\" \
                    }, \
                    { \
                        \"Resource\":[ \
                            \"collection\/tv-*\" \
                        ], \
                        \"Permission\":[ \
                            \"aoss:CreateCollectionItems\" \
                        ], \
                        \"ResourceType\": \"collection\" \
                    } \
                ], \
                \"Principal\":[ \
                    \"arn:aws:iam::123456789012:role\/Admin\" \
                ] \
            }]"
        });
        const response = await client.send(command);
        console.log("Access policy created:");
        console.log(response['accessPolicyDetail']);
    } catch (error) {
        if (error.name === 'ConflictException') {
            console.log('[ConflictException] An access policy with that name already
 exists.');
        } else
            console.error(error);
    };
}

async function createCollection(client) {
    // Creates a collection to hold TV sitcoms indexes
    try {
        var command = new CreateCollectionCommand({
            name: 'tv-sitcoms',
            type: 'SEARCH'
        });
        const response = await client.send(command);
        return (response)
    } catch (error) {
        if (error.name === 'ConflictException') {
            console.log('[ConflictException] A collection with this name already
 exists. Try another name.');
        } else
            console.error(error);
```

```
    };
}

async function waitForCollectionCreation(client) {
    // Waits for the collection to become active
    try {
        var command = new BatchGetCollectionCommand({
            names: ['tv-sitcoms']
        });
        var response = await client.send(command);
        while (response.collectionDetails[0]['status'] == 'CREATING') {
            console.log('Creating collection...')
            await sleep(30000) // Wait for 30 seconds, then check the status again
            function sleep(ms) {
                return new Promise((resolve) => {
                    setTimeout(resolve, ms);
                });
            }
            var response = await client.send(command);
        }
        console.log('Collection successfully created:');
        console.log(response['collectionDetails']);
        // Extract the collection endpoint from the response
        var host = (response.collectionDetails[0]['collectionEndpoint'])
        // Pass collection endpoint to index document request
        indexDocument(host)
    } catch (error) {
        console.error(error);
    };
}

async function indexDocument(host) {

    var client = new Client({
        node: host,
        Connection: class extends Connection {
            buildRequestObject(params) {
                var request = super.buildRequestObject(params)
                request.service = 'aoss';
                request.region = 'us-east-1'; // e.g. us-east-1
                var body = request.body;
                request.body = undefined;
                delete request.headers['content-length'];
                request.headers['x-amz-content-sha256'] = 'UNSIGNED-PAYLOAD';
```

```
                request = aws4.sign(request, AWS.config.credentials);
                request.body = body;

                return request
            }
        }
    });

    // Create an index
    try {
        var index_name = "sitcoms-eighties";

        var response = await client.indices.create({
            index: index_name
        });

        console.log("Creating index:");
        console.log(response.body);

        // Add a document to the index
        var document = "{ \"title\": \"Seinfeld\", \"creator\": \"Larry David\", \"year
\": \"1989\" }\n";

        var response = await client.index({
            index: index_name,
            body: document
        });

        console.log("Adding document:");
        console.log(response.body);
    } catch (error) {
        console.error(error);
    };
}

execute()
```

Uso de CloudFormation para crear colecciones de Amazon OpenSearch sin servidor

Puede utilizar CloudFormation para crear recursos de Amazon OpenSearch sin servidor, como colecciones, políticas de seguridad y puntos de conexión de VPC. Para obtener la referencia

completa de CloudFormation para OpenSearch sin servidor, consulte Amazon OpenSearch sin servidor en la Guía del usuario de CloudFormation.

La siguiente plantilla de ejemplo de CloudFormation crea una política de acceso a datos, una política de red y una política de seguridad simples, así como una colección coincidente. Es una buena forma de empezar a utilizar Amazon OpenSearch sin servidor de forma rápida y de aprovisionar los elementos necesarios para crear y utilizar una colección.

> **Important**
>
> En este ejemplo, se utiliza el acceso a la red pública, que no se recomienda para las cargas de trabajo de producción. Recomendamos utilizar el acceso mediante VPC para proteger las colecciones. Para obtener más información, consulte AWS::OpenSearchServerless::VpcEndpoint y the section called “Puntos de conexión de VPC”.

```
AWSTemplateFormatVersion: 2010-09-09
Description: 'Amazon OpenSearch Serverless template to create an IAM user, encryption
 policy, data access policy and collection'
Resources:
  IAMUSer:
    Type: 'AWS::IAM::User'
    Properties:
      UserName:  aossadmin
  DataAccessPolicy:
    Type: 'AWS::OpenSearchServerless::AccessPolicy'
    Properties:
      Name: quickstart-access-policy
      Type: data
      Description: Access policy for quickstart collection
      Policy: !Sub >-
        [{"Description":"Access for cfn user","Rules":
[{"ResourceType":"index","Resource":["index/*/*"],"Permission":["aoss:*"]},
        {"ResourceType":"collection","Resource":["collection/quickstart"],"Permission":
["aoss:*"]}],
        "Principal":["arn:aws:iam::${AWS::AccountId}:user/aossadmin"]}]
  NetworkPolicy:
    Type: 'AWS::OpenSearchServerless::SecurityPolicy'
    Properties:
      Name: quickstart-network-policy
```

```
      Type: network
      Description: Network policy for quickstart collection
      Policy: >-
        [{"Rules":[{"ResourceType":"collection","Resource":["collection/
quickstart"]}, {"ResourceType":"dashboard","Resource":["collection/
quickstart"]}],"AllowFromPublic":true}]
  EncryptionPolicy:
    Type: 'AWS::OpenSearchServerless::SecurityPolicy'
    Properties:
      Name: quickstart-security-policy
      Type: encryption
      Description: Encryption policy for quickstart collection
      Policy: >-
        {"Rules":[{"ResourceType":"collection","Resource":["collection/
quickstart"]}],"AWSOwnedKey":true}
  Collection:
    Type: 'AWS::OpenSearchServerless::Collection'
    Properties:
      Name: quickstart
      Type: TIMESERIES
      Description: Collection to holds timeseries data
    DependsOn: EncryptionPolicy
Outputs:
  IAMUser:
    Value: !Ref IAMUSer
  DashboardURL:
    Value: !GetAtt Collection.DashboardEndpoint
  CollectionARN:
    Value: !GetAtt Collection.Arn
```

Hacer copias de seguridad de las colecciones mediante instantáneas

Las instantáneas son una copias de seguridad puntuales de sus colecciones sin Amazon OpenSearch sin servidor que ofrecen funciones de recuperación ante desastres. OpenSearch sin servidor crea y administra automáticamente instantáneas de sus colecciones, a fin de garantizar la continuidad empresarial y la protección de los datos. Cada instantánea contiene:

- Metadatos del índice: Configuración y asignaciones de sus índices.
- Metadatos del clúster: Plantillas de índice y alias.
- Datos del índice: Todos los documentos y datos almacenados en sus índices.

Ventajas principales

- Copias de seguridad automáticas cada hora sin necesidad de configuración manual
- Gastos generales de mantenimiento cero
- Sin costos de almacenamiento adicionales
- Recuperación rápida en caso de pérdida accidental de datos
- Posibilidad de restaurar índices específicos a partir de una instantánea

Consideraciones importantes

- Crear una instantánea no es instantáneo y requiere tiempo para completarse.
- Los documentos nuevos o las actualizaciones que se produzcan durante la creación de la instantánea podrían no incluirse en la instantánea.
- Solo puede restaurar instantáneas en su colección original y no en una nueva.
- Al restaurarse, los índices reciben nuevos UUID que difieren de sus versiones originales.
- Solo puede ejecutar una operación de restauración a la vez.
- No puede iniciar varias operaciones de restauración en la misma colección al mismo tiempo. Intentar restaurar índices durante una operación de restauración activa hace que la operación falle.
- Durante una operación de restauración, sus solicitudes a los índices fallan.

Permisos necesarios

Para trabajar con instantáneas, configure los siguientes permisos en su política de acceso a datos. Para obtener más información acerca de las políticas de acceso a datos, consulte Políticas de acceso a datos frente a políticas de IAM.

Política de acceso a los datos	API
aoss:DescribeSnapshot	GET /_cat/snapshots GET /_cat/snapshots/aoss-automated/ GET _snapshot/aoss-automated/*snapshot*/
aoss:RestoreSnapshot	POST /_snapshot/aoss-automated/*snapshot*/_restore

Política de acceso a los datos	API
aoss:DescribeCollectionItems	GET /_cat/recovery

Puede configurar las políticas usando los siguientes comandos de AWS CLI:

1. create-access-policy
2. delete-access-policy
3. get-access-policy
4. update-access-policy

A continuación, se muestra un ejemplo de un comando de la CLI para crear una política de acceso:

```
aws opensearchserverless create-access-policy \
--type data \
--name AWSExample-data-access-policy \
--region us-west-2 \
--policy '[
  {
    "Rules": [
      {
        "Resource": [
          "collection/AWSExample-collection"
        ],
        "Permission": [
          "aoss:DescribeSnapshot",
          "aoss:RestoreSnapshot",
          "aoss:DescribeCollectionItems"
        ],
        "ResourceType": "collection"
      }
    ],
    "Principal": [
      "arn:aws:iam::AWSExample-account-ID:user/AWSExample-user"
    ],
    "Description": "Data policy to support snapshot operations."
  }
]'
```

Uso de instantáneas

De manera predeterminada, cuando usted crea una nueva colección, OpenSearch sin servidor crea instantáneas automáticamente cada hora. No tiene que hacer nada. Cada instantánea incluye todos los índices de la colección. Después de que OpenSearch sin servidor crea las instantáneas, usted puede enumerarlas y ver los detalles de cada instantánea mediante los siguientes comandos.

Enumerar instantáneas

Utilice el siguiente comando para enumerar todas las instantáneas de una colección.

```
GET /_cat/snapshots/aoss-automated/
```

OpenSearch sin servidor devuelve una respuesta similar a la siguiente:

```
id                                status  start_epoch start_time end_epoch  end_time
  duration    indices successful_shards failed_shards total_shards
snapshot-AWSExampleSnapshotID1     SUCCESS 1737964331  07:52:11   1737964382 07:53:02
  50.4s       1
snapshot-AWSExampleSnapshotID2     SUCCESS 1737967931  08:52:11   1737967979 08:52:59
  47.7s       2
snapshot-AWSExampleSnapshotID3     SUCCESS 1737971531  09:52:11   1737971581 09:53:01
  49.1s       3
snapshot-AWSExampleSnapshotID4 IN_PROGRESS 1737975131  10:52:11   -          -
   4.8d       3
```

Obtener instantáneas

Obtiene información sobre una instantánea.

```
GET _snapshot/aoss-automated/snapshot/
```

Solicitud de ejemplo

```
GET _snapshot/aoss-automated/snapshot-AWSExampleSnapshotID1/
```

Respuesta de ejemplo

```
{
```

```
    "snapshots": [
        {
            "snapshot": "snapshot-AWSExampleSnapshotID1-5e01-4423-9833Example",
            "uuid": "AWSExample-5e01-4423-9833-9e9eb757Example",
            "version_id": 136327827,
            "version": "2.11.0",
            "remote_store_index_shallow_copy": true,
            "indices": [
                "AWSExample-index-0117"
            ],
            "data_streams": [],
            "include_global_state": true,
            "metadata": {},
            "state": "SUCCESS",
            "start_time": "2025-01-27T09:52:11.953Z",
            "start_time_in_millis": 1737971531953,
            "end_time": "2025-01-27T09:53:01.062Z",
            "end_time_in_millis": 1737971581062,
            "duration_in_millis": 49109,
            "failures": [],
            "shards": {
                "total": 0,
                "failed": 0,
                "successful": 0
            }
        }
    ]
}
```


Comprensión de los campos de la respuesta de una instantánea

id

Un identificador único de la operación de la instantánea.

status

El estado actual de la operación de instantáneas. Los valores posibles son:

- SUCCESS
- IN_PROGRESS

duración

El tiempo que lleva completar la operación de instantáneas.

indices

El número de índices incluidos en la instantánea.

Restauración a partir de una instantánea

La restauración a partir de una instantánea permite recuperar los datos de una copia de seguridad realizada anteriormente. Este proceso es crucial para la recuperación ante desastres y la administración de datos en OpenSearch sin servidor.

Consideraciones importantes

1. Los índices restaurados tendrán UUID diferentes a los de sus versiones originales.
2. Las instantáneas solo se pueden restaurar a su colección original. No se admite la restauración entre colecciones.
3. Las operaciones de restauración pueden afectar al rendimiento del clúster. Planifíquelo en consecuencia.

Para restaurar los incidentes respaldados a partir de una instantánea

1. Ejecute el siguiente comando para identificar la instantánea adecuada.

   ```
   GET /_snapshot/aoss-automated/_all
   ```

   Para obtener una lista de instantáneas más pequeña, ejecute el siguiente comando.

   ```
   GET /_cat/snapshots/aoss-automated/
   ```

2. Ejecute el siguiente comando para verificar los detalles de la instantánea antes de restaurar.

   ```
   GET _snapshot/aoss-automated/snapshot-AWSExampleSnapshotID1/
   ```

3. Ejecute el siguiente comando para restaurar a partir de una instantánea específica.

   ```
   POST /_snapshot/aoss-automated/snapshot-ID/_restore
   ```

   Puede personalizar la operación de restauración incluyendo el cuerpo de la solicitud. A continuación se muestra un ejemplo.

```
POST /_snapshot/aoss-automated/snapshot-
AWSExampleSnapshotID1-5e01-4423-9833Example/_restore
{
  "indices": "opensearch-dashboards*,my-index*",
  "ignore_unavailable": true,
  "include_global_state": false,
  "include_aliases": false,
  "rename_pattern": "opensearch-dashboards(.+)",
  "rename_replacement": "restored-opensearch-dashboards$1"
}
```

4. Ejecute el siguiente comando para ver el progreso de la restauración.

```
GET /_cat/recovery
```

> **Note**
>
> Al restaurar una instantánea con un comando que incluye el cuerpo de la solicitud, puede usar los siguientes parámetros para controlar el comportamiento de la restauración:
>
> indices
>
> Especifica qué índices se van a restaurar. Este parámetro admite patrones comodín.
>
> ignore_unavailable
>
> Permite que la operación de restauración continúe incluso si falta un índice en la instantánea.
>
> include_global_state
>
> Determina si se debe restaurar el estado del clúster.
>
> include_aliases
>
> Controla si se restauran los alias asociados.
>
> rename_pattern y rename_replacement
>
> Le permite cambiar el nombre de los índices durante la operación de restauración.

Administración de los límites de capacidad de Amazon OpenSearch sin servidor

Con Amazon OpenSearch sin servidor, no tiene que administrar la capacidad usted mismo. OpenSearch sin servidor escala automáticamente la capacidad de cómputo de su cuenta según la carga de trabajo actual. La capacidad de procesamiento sin servidor se mide en las unidades de cómputo (OCU) de OpenSearch. Cada OCU es una combinación de 6 GiB de memoria y la CPU virtual (vCPU) correspondiente, así como la transferencia de datos a Amazon S3. Para obtener más información sobre la arquitectura desacoplada de OpenSearch sin servidor, consulte the section called “Funcionamiento”.

Cuando usted crea su primera colección, OpenSearch sin servidor inicia OCUs según su configuración de redundancia. De manera predeterminada, las réplicas activas redundantes están habilitadas, lo que significa que se inician cuatro OCUs en total (dos para indexación y dos para búsqueda) para garantizar alta disponibilidad con nodos en espera en otra zona de disponibilidad. Para fines de desarrollo y prueba, usted puede inhabilitar la opción Habilitar redundancia para una colección, lo cual elimina las réplicas en espera e inicia únicamente dos OCU (una para indexación y una para búsqueda). Siempre existen estas OCU, incluso cuando no hay actividad de indexación o búsqueda. Todas las colecciones posteriores pueden compartir estas OCU (excepto las colecciones con claves de aislamiento únicas de AWS KMS, que inician su propio conjunto de OCU). Si es necesario, OpenSearch sin servidor escala horizontalmente de manera automática y agrega OCU adicionales a medida que aumenta el uso de indexación y búsqueda. Cuando el tráfico en el punto de conexión de la colección disminuye, la capacidad vuelve a reducirse verticalmente hasta la cantidad mínima de OCU requerida para el tamaño de los datos. Para las colecciones de búsqueda y series temporales, la cantidad de OCUs necesarias cuando están inactivas es proporcional al tamaño de los datos y a la cantidad de índices. Para vectores, depende tanto de la memoria RAM necesaria para almacenar los gráficos vectoriales como del espacio en disco necesario para almacenar los índices. Si no está en un estado inactivo, los requisitos de OCU tienen en cuenta ambos factores.

Las colecciones de vectores mantienen los datos de índices en el almacenamiento local de la OCU. Los límites de RAM de la OCU se alcanzan más rápido que los límites de disco de la OCU, lo que hace que las colecciones de vectores estén limitadas por el espacio de RAM. Con la redundancia habilitada, la capacidad de OCU se reduce verticalmente a un mínimo de 1 OCU [0,5 OCU × 2] para la indexación y 1 OCU [0,5 OCU × 2] para la búsqueda. Cuando deshabilita la redundancia, su dominio se puede reducir verticalmente a 0,5 OCU para la indexación y 0,5 OCU para la búsqueda. La escalabilidad también tiene en cuenta la cantidad de particiones necesarias para su colección o índice. Cada OCU puede soportar un número específico de particiones. El número de índices debe

ser proporcional al número de particiones. El número total de OCU base requeridos es la cantidad máxima necesaria según sus datos, memoria y particiones. Para obtener más información, consulte Capacidades de búsqueda rentables de Amazon OpenSearch sin servidor, a cualquier escala, en el Blog de macrodatos de AWS.

En el caso de las colecciones de búsqueda y de búsquedas vectoriales, todos los datos se almacenan en índices calientes para garantizar tiempos de respuesta rápidos a las consultas. Las colecciones de series temporales utilizan una combinación de almacenamiento en caliente y templado, lo que almacena los datos más recientes en un almacenamiento en caliente para optimizar los tiempos de respuesta a las consultas para los datos a los que se accede con mayor frecuencia. Para obtener más información, consulte the section called “Elección de un tipo de colección”.

> **Note**
>
> Una colección de búsqueda vectorial no puede compartir OCU con colecciones de búsqueda y serie temporal, incluso si la colección de búsqueda vectorial use la misma clave de KMS que las colecciones de búsqueda o serie temporal. Se creará un nuevo conjunto de OCU para su primera colección de vectores. Las OCU de las colecciones de vectores se comparten entre las mismas colecciones de claves de KMS.

Para administrar la capacidad de las colecciones y controlar los costos, puede especificar la capacidad máxima general de indexación y búsqueda para la cuenta y región actuales, y OpenSearch sin servidor escala horizontalmente los recursos de las colecciones de manera automática en función de estas especificaciones.

Como la capacidad de indexación y búsqueda se escalan por separado, debe especificar los límites de cuenta para cada una de ellas:

- Capacidad máxima de indexación: OpenSearch sin servidor puede aumentar la capacidad de indexación hasta este número de OCU.
- Capacidad máxima de búsqueda: OpenSearch sin servidor puede aumentar la capacidad de indexación hasta este número de OCU.

> **Note**
>
> En este momento, la configuración de capacidad solo se aplica según el nivel de cuenta. No puede configurar los límites de capacidad por colección.

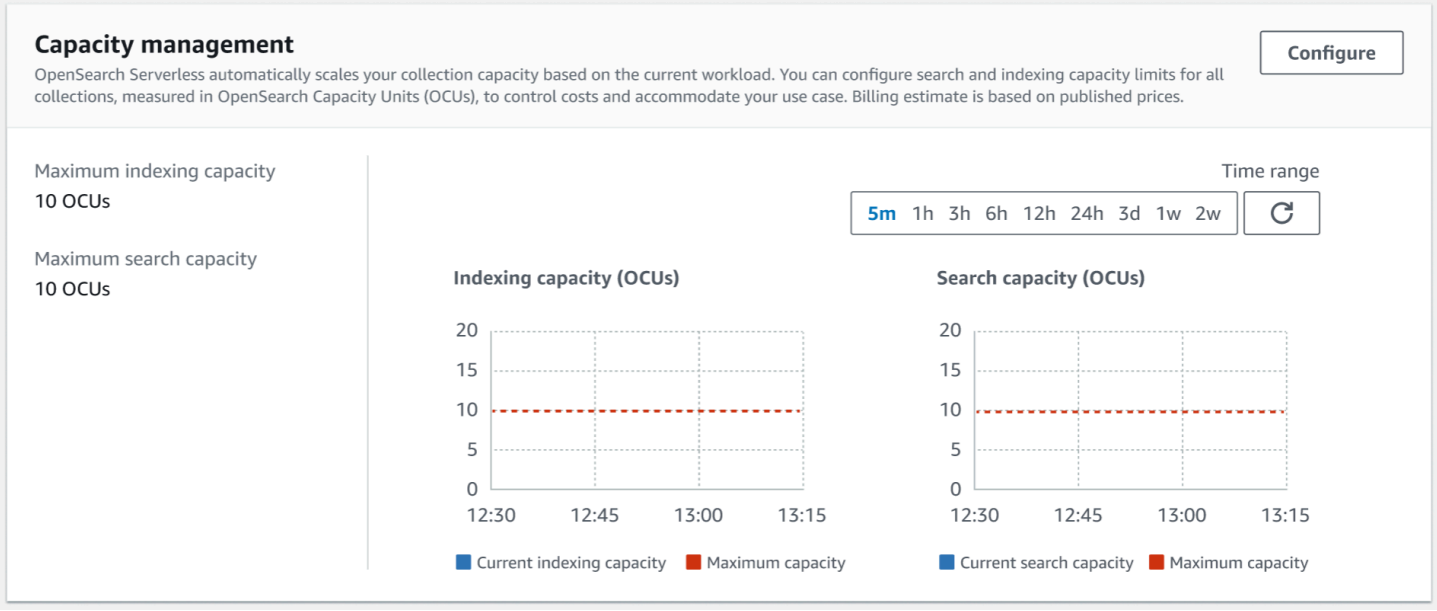
El objetivo debe ser garantizar que la capacidad máxima sea lo bastante alta como para asumir los picos de carga de trabajo. Según la configuración, OpenSearch sin servidor escala horizontalmente de manera automática la cantidad de OCU de las colecciones para procesar la carga de trabajo de indexación y búsqueda.

Temas

- Como establecer los parámetros de capacidad
- Límites de la capacidad máxima
- Monitoreo del uso de la capacidad

Como establecer los parámetros de capacidad

Para configurar los ajustes de capacidad en la consola de OpenSearch sin servidor, expanda Sin servidor en el panel de navegación izquierdo y seleccione Panel de control. Especifique la capacidad máxima de indexación y búsqueda en Administración de capacidades:



Para configurar la capacidad mediante la AWS CLI, envíe una solicitud updateAccountSettings:

```
aws opensearchserverless update-account-settings \
    --capacity-limits '{ "maxIndexingCapacityInOCU": 8,"maxSearchCapacityInOCU": 9 }'
```

Límites de la capacidad máxima

El total máximo de índices que puede contener una colección es 1000. Para los tres tipos de colecciones, la capacidad máxima de OCU predeterminada es de 10 OCUs para indexación y 10 OCUs para búsqueda. La capacidad mínima de OCU permitida para una cuenta es de 1 OCU [0,5 OCU x 2] para indexación y 1 OCU [0,5 OCU x 2] para búsqueda. Para todas las colecciones, la capacidad máxima permitida es de 1,700 OCU para la indexación y 1,700 OCU para la búsqueda. Puede configurar el recuento de OCU para que sea cualquier número desde 1 hasta la capacidad máxima permitida, en múltiplos de 2.

Cada OCU incluye suficiente almacenamiento efímero en caliente para 120 GiB de datos de índice. OpenSearch sin servidor admite hasta 1 TiB de datos por índice en colecciones de búsqueda y búsquedas vectoriales y 100 TiB de datos recientes por índice en una colección de series temporales. En el caso de las recopilaciones de series temporales, puede incorporar aún más datos, que se pueden almacenar como datos templados en S3.

Para ver una lista de todas las cuotas, consulte Cuotas de OpenSearch sin servidor.

Monitoreo del uso de la capacidad

Puede monitorear las métricas de CloudWatch `Search0CU` y `Indexing0CU` a nivel de cuenta para comprender cómo se escalan sus colecciones. Le recomendamos que configure alarmas para que le notifiquen si su cuenta se está acercando a un límite para las métricas relacionadas con la capacidad, de modo que pueda ajustar su configuración de capacidad en consecuencia.

Además, puede utilizar estas métricas para determinar si su configuración de capacidad máxima es adecuada o si necesita ajustarla. Analice estas métricas para centrar sus esfuerzos para optimizar la eficiencia de sus colecciones. Para obtener más información sobre las métricas que OpenSearch sin servidor envía a CloudWatch, consulte the section called “Monitoreo de OpenSearch sin servidor”.

Ingesta de datos en las colecciones de Amazon OpenSearch sin servidor

Estas secciones proporcionan detalles sobre las canalizaciones de ingesta compatibles para la ingesta de datos en las colecciones de Amazon OpenSearch sin servidor. También describen

algunos de los clientes que puede utilizar para interactuar con las operaciones de la API de OpenSearch. Sus clientes deben ser compatibles con OpenSearch 2.x para poder integrarse con OpenSearch sin servidor.

Temas

- Permisos mínimos necesarios
- OpenSearch Ingestion
- Fluent Bit
- Amazon Data Firehose
- Go
- Java
- JavaScript
- Logstash
- Python
- Ruby
- Firma de solicitudes HTTP con otros clientes

Permisos mínimos necesarios

Para poder incorporar datos a una colección de OpenSearch sin servidor, la entidad principal que escribe los datos debe tener los siguientes permisos mínimos asignados en una política de acceso a los datos:

```
[
   {
      "Rules":[
         {
            "ResourceType":"index",
            "Resource":[
               "index/target-collection/logs"
            ],
            "Permission":[
               "aoss:CreateIndex",
               "aoss:WriteDocument",
               "aoss:UpdateIndex"
            ]
         }
```

```
      ],
      "Principal":[
         "arn:aws:iam::123456789012:user/my-user"
      ]
   }
]
```

Los permisos pueden ser más amplios si planea escribir en índices adicionales. Por ejemplo, en lugar de especificar un único índice de destino, puede conceder permisos a todos los índices (index/*target-collection*/*) o a un subconjunto de índices (index/*target-collection*/*logs**).

Para obtener una referencia de todas las operaciones de la API de OpenSearch disponibles y sus permisos asociados, consulte the section called “Operaciones y complementos compatibles”.

OpenSearch Ingestion

En lugar de utilizar un cliente de terceros para enviar datos directamente a una colección de OpenSearch sin servidor, puede utilizar Amazon OpenSearch Ingestion. Puede configurar los productores de datos para que envíen datos a OpenSearch Ingestion, que a su vez los envían automáticamente a la colección que especifique. También puede configurar OpenSearch Ingestion para transformar los datos antes de entregarlos. Para más información, consulte Amazon OpenSearch Ingestion.

Una canalización de OpenSearch Ingestion necesita permiso para escribir en una colección de OpenSearch sin servidor que esté configurada como su receptor. Estos permisos incluyen la capacidad de describir la colección y enviarle solicitudes HTTP. Para obtener instrucciones sobre cómo usar OpenSearch Ingestion para agregar datos a una colección, consulte the section called “Concesión de acceso a colecciones para canalizaciones”.

Para comenzar con OpenSearch Ingestion, consulte the section called “Tutorial: incorporar datos en una colección”.

Fluent Bit

Puede utilizar la imagen de AWS for Fluent Bit y el complemento de salida de OpenSearch para realizar la ingesta de datos en colecciones de OpenSearch sin servidor.

> **Note**
>
> Debe tener la versión 2.30.0 o posterior de la imagen de AWS for Fluent Bit para poder integrarla con OpenSearch sin servidor.

Ejemplo de configuración:

En este ejemplo de sección de resultados del archivo de configuración se muestra cómo utilizar una colección de OpenSearch sin servidor como destino. La adición importante es el parámetro `AWS_Service_Name`, que es `aoss`. `Host` es el punto de conexión de la colección.

```
[OUTPUT]
    Name  opensearch
    Match *
    Host  collection-endpoint.us-west-2.aoss.amazonaws.com
    Port  443
    Index  my_index
    Trace_Error On
    Trace_Output On
    AWS_Auth On
    AWS_Region <region>
    AWS_Service_Name aoss
    tls     On
    Suppress_Type_Name On
```

Amazon Data Firehose

Firehose es compatible con OpenSearch sin servidor como destino de entrega. Para obtener instrucciones para enviar los datos a OpenSearch sin servidor, consulte Creating a Kinesis Data Firehose Delivery Stream y Choose OpenSearch Serverless for Your Destination en la Guía para desarrolladores de Amazon Data Firehose.

El rol de IAM que proporcione a Firehose para la entrega se debe especificar dentro de una política de acceso a datos con el permiso mínimo `aoss:WriteDocument` para la colección de destino y debe tener un índice preexistente al que enviar datos. Para obtener más información, consulte the section called “Permisos mínimos necesarios”.

Es posible que necesite realizar transformaciones en los datos antes de enviarlos a OpenSearch sin servidor. Para obtener más información acerca de cómo utilizar las funciones de Lambda para realizar esta tarea, consulte Amazon Kinesis Data Firehose Data Transformation en la misma guía.

Go

En el siguiente ejemplo de código se utiliza el cliente opensearch-go para Go con el fin de establecer una conexión segura con la colección de OpenSearch sin servidor especificada y crear un índice único. Debe proporcionar valores para `region` y `host`.

```
package main

import (
  "context"
  "log"
  "strings"
  "github.com/aws/aws-sdk-go-v2/aws"
  "github.com/aws/aws-sdk-go-v2/config"
  opensearch "github.com/opensearch-project/opensearch-go/v2"
  opensearchapi "github.com/opensearch-project/opensearch-go/v2/opensearchapi"
  requestsigner "github.com/opensearch-project/opensearch-go/v2/signer/awsv2"
)

const endpoint = "" // serverless collection endpoint

func main() {
 ctx := context.Background()

 awsCfg, err := config.LoadDefaultConfig(ctx,
  config.WithRegion("<AWS_REGION>"),
  config.WithCredentialsProvider(
   getCredentialProvider("<AWS_ACCESS_KEY>", "<AWS_SECRET_ACCESS_KEY>",
 "<AWS_SESSION_TOKEN>"),
  ),
 )
 if err != nil {
  log.Fatal(err) // don't log.fatal in a production-ready app
 }

 // create an AWS request Signer and load AWS configuration using default config folder
 or env vars.
 signer, err := requestsigner.NewSignerWithService(awsCfg, "aoss") // "aoss" for Amazon
 OpenSearch Serverless
```

```
if err != nil {
 log.Fatal(err) // don't log.fatal in a production-ready app
}

// create an opensearch client and use the request-signer
client, err := opensearch.NewClient(opensearch.Config{
 Addresses: []string{endpoint},
 Signer:    signer,
})
if err != nil {
 log.Fatal("client creation err", err)
}

indexName := "go-test-index"

 // define index mapping
mapping := strings.NewReader(`{
 "settings": {
   "index": {
        "number_of_shards": 4
        }
      }
 }`)

// create an index
createIndex := opensearchapi.IndicesCreateRequest{
 Index: indexName,
   Body: mapping,
}
createIndexResponse, err := createIndex.Do(context.Background(), client)
if err != nil {
 log.Println("Error ", err.Error())
 log.Println("failed to create index ", err)
 log.Fatal("create response body read err", err)
}
log.Println(createIndexResponse)

// delete the index
deleteIndex := opensearchapi.IndicesDeleteRequest{
 Index: []string{indexName},
}

deleteIndexResponse, err := deleteIndex.Do(context.Background(), client)
if err != nil {
```

```
  log.Println("failed to delete index ", err)
  log.Fatal("delete index response body read err", err)
 }
 log.Println("deleting index", deleteIndexResponse)
}

func getCredentialProvider(accessKey, secretAccessKey, token string)
 aws.CredentialsProviderFunc {
 return func(ctx context.Context) (aws.Credentials, error) {
  c := &aws.Credentials{
   AccessKeyID:     accessKey,
   SecretAccessKey: secretAccessKey,
   SessionToken:    token,
  }
  return *c, nil
 }
}
```

Java

En el siguiente ejemplo de código se utiliza el cliente opensearch-java para Java con el fin de establecer una conexión segura con la colección de OpenSearch sin servidor especificada y crear un índice único. Debe proporcionar valores para `region` y `host`.

La diferencia importante con respecto a los dominios de OpenSearch Service es el nombre del servicio (`aoss` en lugar de `es`).

```
// import OpenSearchClient to establish connection to OpenSearch Serverless collection
import org.opensearch.client.opensearch.OpenSearchClient;

SdkHttpClient httpClient = ApacheHttpClient.builder().build();
// create an opensearch client and use the request-signer
OpenSearchClient client = new OpenSearchClient(
    new AwsSdk2Transport(
        httpClient,
        "...us-west-2.aoss.amazonaws.com", // serverless collection endpoint
        "aoss" // signing service name
        Region.US_WEST_2, // signing service region
        AwsSdk2TransportOptions.builder().build()
    )
);

String index = "sample-index";
```

```
// create an index
CreateIndexRequest createIndexRequest = new
 CreateIndexRequest.Builder().index(index).build();
CreateIndexResponse createIndexResponse = client.indices().create(createIndexRequest);
System.out.println("Create index reponse: " + createIndexResponse);

// delete the index
DeleteIndexRequest deleteIndexRequest = new
 DeleteIndexRequest.Builder().index(index).build();
DeleteIndexResponse deleteIndexResponse = client.indices().delete(deleteIndexRequest);
System.out.println("Delete index reponse: " + deleteIndexResponse);

httpClient.close();
```

El siguiente código de ejemplo vuelve a establecer una conexión segura y, a continuación, busca en un índice.

```
import org.opensearch.client.opensearch.OpenSearchClient;

SdkHttpClient httpClient = ApacheHttpClient.builder().build();

OpenSearchClient client = new OpenSearchClient(
    new AwsSdk2Transport(
        httpClient,
        "...us-west-2.aoss.amazonaws.com", // serverless collection endpoint
        "aoss" // signing service name
        Region.US_WEST_2, // signing service region
        AwsSdk2TransportOptions.builder().build()
    )
);

Response response = client.generic()
    .execute(
        Requests.builder()
            .endpoint("/" + "users" + "/_search?typed_keys=true")
            .method("GET")
            .json("{"
                + "    \"query\": {"
                + "        \"match_all\": {}"
                + "    }"
                + "}")
            .build());
```

```
httpClient.close();
```

JavaScript

En el siguiente ejemplo de código se utiliza el cliente opensearch-js para JavaScript con el fin de establecer una conexión segura con la colección de OpenSearch sin servidor especificada, crear un índice único, agregar un documento y eliminar el índice. Debe proporcionar valores para `node` y `region`.

La diferencia importante con respecto a los dominios de OpenSearch Service es el nombre del servicio (`aoss` en lugar de `es`).

Version 3

En este ejemplo, se utiliza la versión 3 del SDK para JavaScript en Node.js.

```
const { defaultProvider } = require('@aws-sdk/credential-provider-node');
const { Client } = require('@opensearch-project/opensearch');
const { AwsSigv4Signer } = require('@opensearch-project/opensearch/aws');

async function main() {
    // create an opensearch client and use the request-signer
    const client = new Client({
        ...AwsSigv4Signer({
            region: 'us-west-2',
            service: 'aoss',
            getCredentials: () => {
                const credentialsProvider = defaultProvider();
                return credentialsProvider();
            },
        }),
        node: '' # // serverless collection endpoint
    });

    const index = 'movies';

    // create index if it doesn't already exist
    if (!(await client.indices.exists({ index })).body) {
        console.log((await client.indices.create({ index })).body);
    }

    // add a document to the index
```

```
    const document = { foo: 'bar' };
    const response = await client.index({
        id: '1',
        index: index,
        body: document,
    });
    console.log(response.body);

    // delete the index
    console.log((await client.indices.delete({ index })).body);
}

main();
```

Version 2

En este ejemplo, se utiliza la versión 2 del SDK para JavaScript en Node.js.

```
const AWS = require('aws-sdk');
const { Client } = require('@opensearch-project/opensearch');
const { AwsSigv4Signer } = require('@opensearch-project/opensearch/aws');

async function main() {
    // create an opensearch client and use the request-signer
    const client = new Client({
        ...AwsSigv4Signer({
            region: 'us-west-2',
            service: 'aoss',
            getCredentials: () =>
                new Promise((resolve, reject) => {
                    AWS.config.getCredentials((err, credentials) => {
                        if (err) {
                            reject(err);
                        } else {
                            resolve(credentials);
                        }
                    });
                }),
        }),
        node: '' # // serverless collection endpoint
    });

    const index = 'movies';
```

```
    // create index if it doesn't already exist
    if (!(await client.indices.exists({ index })).body) {
        console.log((await client.indices.create({
            index
        })).body);
    }

    // add a document to the index
    const document = {
        foo: 'bar'
    };
    const response = await client.index({
        id: '1',
        index: index,
        body: document,
    });
    console.log(response.body);

    // delete the index
    console.log((await client.indices.delete({ index })).body);
}

main();
```

Logstash

Puede usar el complemento Logstash OpenSearch para publicar registros en las colecciones de OpenSearch sin servidor.

Cómo usar Logstash para enviar datos a OpenSearch sin servidor

1. Instale la versión 2.0.0 o posterior del complemento logstash-output-opensearch con Docker o Linux.

   Docker

   Docker aloja el software de Logstash OSS con el complemento de salida de OpenSearch preinstalado: opensearchproject/logstash-oss-with-opensearch-output-plugin. Puede extraer la imagen como cualquier otra:

   ```
   docker pull opensearchproject/logstash-oss-with-opensearch-output-plugin:latest
   ```

Linux

En primer lugar, instale la última versión de Logstash si aún no lo hizo. A continuación, instale la versión 2.0.0 del complemento de salida:

```
cd logstash-8.5.0/
bin/logstash-plugin install --version 2.0.0 logstash-output-opensearch
```

Si el complemento ya está instalado, actualícelo a la versión más reciente:

```
bin/logstash-plugin update logstash-output-opensearch
```

A partir de la versión 2.0.0 del complemento, el SDK de AWS usa la versión 3. Si utiliza una versión de Logstash anterior a la 8.4.0, debe eliminar todos los complementos de AWS preinstalados e instalar el complemento `logstash-integration-aws`:

```
/usr/share/logstash/bin/logstash-plugin remove logstash-input-s3
/usr/share/logstash/bin/logstash-plugin remove logstash-input-sqs
/usr/share/logstash/bin/logstash-plugin remove logstash-output-s3
/usr/share/logstash/bin/logstash-plugin remove logstash-output-sns
/usr/share/logstash/bin/logstash-plugin remove logstash-output-sqs
/usr/share/logstash/bin/logstash-plugin remove logstash-output-cloudwatch

/usr/share/logstash/bin/logstash-plugin install --version 0.1.0.pre logstash-
integration-aws
```

2. Para que el complemento de salida de OpenSearch funcione con OpenSearch sin servidor, debe realizar las siguientes modificaciones en la sección de salida de `opensearch` de logstash.conf:

   - Especifique `aoss` como el `service_name` en `auth_type`.
   - Especifique el punto de conexión de la colección para `hosts`.
   - Agregue los parámetros `default_server_major_version` y `legacy_template`. Estos parámetros son necesarios para que el complemento funcione con OpenSearch sin servidor.

```
output {
  opensearch {
    hosts => "collection-endpoint:443"
    auth_type => {
```

```
      ...
      service_name => 'aoss'
    }
    default_server_major_version => 2
    legacy_template => false
  }
}
```

Este archivo de configuración de ejemplo toma su entrada desde los archivos en un bucket de S3 y los envía a una colección OpenSearch sin servidor:

```
input {
  s3  {
    bucket => "my-s3-bucket"
    region => "us-east-1"
  }
}

output {
  opensearch {
    ecs_compatibility => disabled
    hosts => "https://my-collection-endpoint.us-east-1.aoss.amazonaws.com:443"
    index => my-index
    auth_type => {
      type => 'aws_iam'
      aws_access_key_id => 'your-access-key'
      aws_secret_access_key => 'your-secret-key'
      region => 'us-east-1'
      service_name => 'aoss'
    }
    default_server_major_version => 2
    legacy_template => false
  }
}
```

3. A continuación, ejecute Logstash con la nueva configuración para probar el complemento:

```
bin/logstash -f config/test-plugin.conf
```

Python

En el siguiente ejemplo de código se utiliza el cliente opensearch-py para Python con el fin de establecer una conexión segura con la colección de OpenSearch sin servidor especificada, crear un índice único y buscar en él. Debe proporcionar valores para `region` y `host`.

La diferencia importante con respecto a los dominios de OpenSearch Service es el nombre del servicio (`aoss` en lugar de `es`).

```
from opensearchpy import OpenSearch, RequestsHttpConnection, AWSV4SignerAuth
import boto3

host = ''  # serverless collection endpoint, without https://
region = ''  # e.g. us-east-1

service = 'aoss'
credentials = boto3.Session().get_credentials()
auth = AWSV4SignerAuth(credentials, region, service)

# create an opensearch client and use the request-signer
client = OpenSearch(
    hosts=[{'host': host, 'port': 443}],
    http_auth=auth,
    use_ssl=True,
    verify_certs=True,
    connection_class=RequestsHttpConnection,
    pool_maxsize=20,
)

# create an index
index_name = 'books-index'
create_response = client.indices.create(
    index_name
)

print('\nCreating index:')
print(create_response)

# index a document
document = {
  'title': 'The Green Mile',
  'director': 'Stephen King',
  'year': '1996'
```

```
}

response = client.index(
    index = 'books-index',
    body = document,
    id = '1'
)


# delete the index
delete_response = client.indices.delete(
    index_name
)

print('\nDeleting index:')
print(delete_response)
```

Ruby

La gema `opensearch-aws-sigv4` proporciona acceso a OpenSearch sin servidor y OpenSearch Service directamente. Dispone de todas las funciones del cliente opensearch-ruby, ya que es una dependencia de esta gema.

Cuando cree una instancia del firmante de Sigv4, especifique `aoss` como nombre del servicio:

```
require 'opensearch-aws-sigv4'
require 'aws-sigv4'

signer = Aws::Sigv4::Signer.new(service: 'aoss',
                                region: 'us-west-2',
                                access_key_id: 'key_id',
                                secret_access_key: 'secret')

# create an opensearch client and use the request-signer
client = OpenSearch::Aws::Sigv4Client.new(
  { host: 'https://your.amz-opensearch-serverless.endpoint',
    log: true },
  signer)

# create an index
index = 'prime'
client.indices.create(index: index)
```

```
# insert data
client.index(index: index, id: '1', body: { name: 'Amazon Echo',
                                            msrp: '5999',
                                            year: 2011 })

# query the index
client.search(body: { query: { match: { name: 'Echo' } } })

# delete index entry
client.delete(index: index, id: '1')

# delete the index
client.indices.delete(index: index)
```

Firma de solicitudes HTTP con otros clientes

Se aplican los siguientes requisitos para firmar solicitudes de colecciones de OpenSearch sin servidor cuando se crean solicitudes HTTP con otros clientes.

- Debe especificar el nombre del servicio como `aoss`.
- El encabezado `x-amz-content-sha256` es obligatorio para todas las solicitudes de Signature Version 4 de AWS. Proporciona un hash de la carga de solicitud. Si hay una carga de solicitud, establezca el valor en su hash criptográfico del algoritmo de hash seguro (SHA) (SHA256). Si no hay ninguna carga de solicitud, establezca el valor en `e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855`, que es el hash de una cadena vacía.

Temas

- Indexación con cURL
- Indexación con Postman

Indexación con cURL

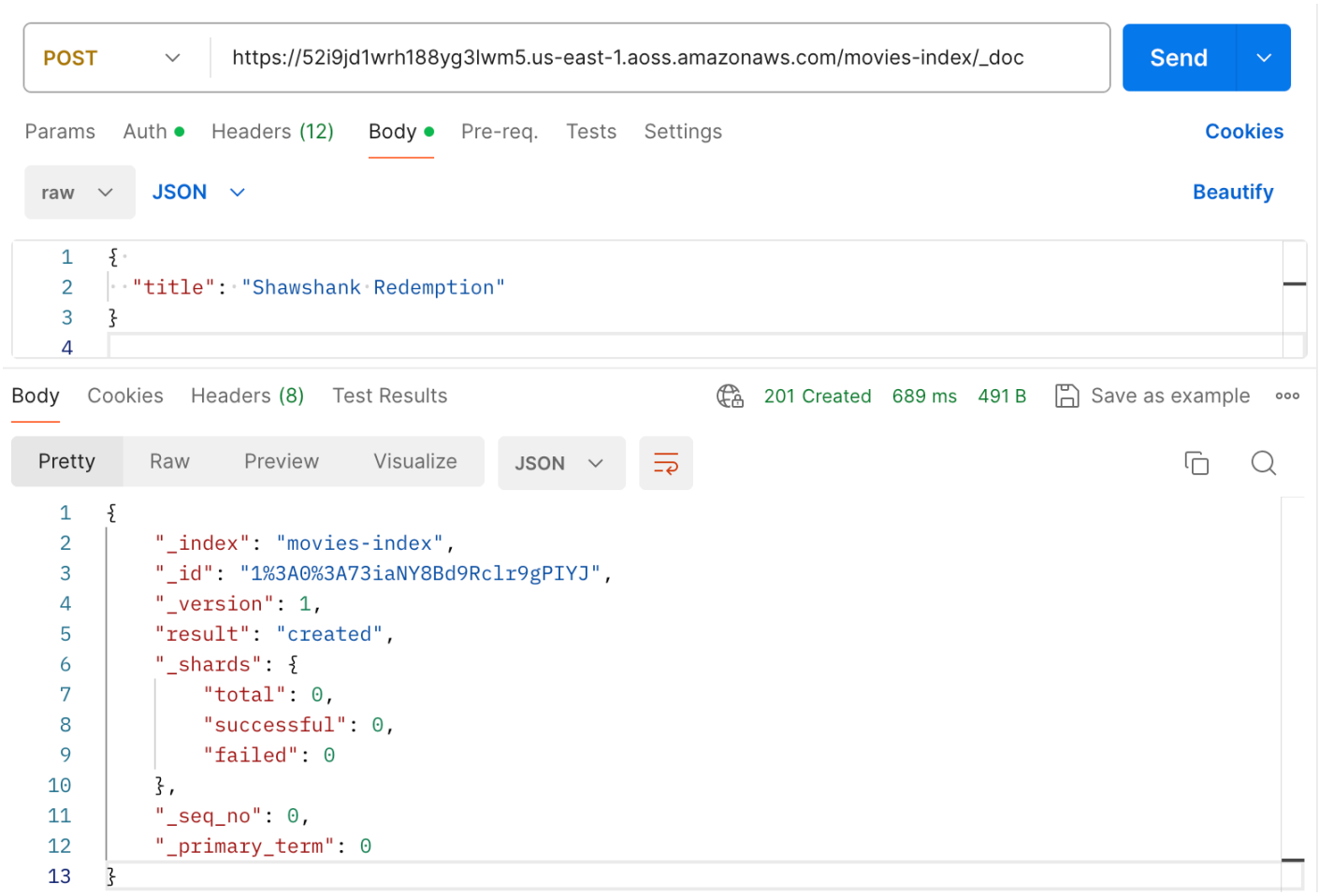
La siguiente solicitud de ejemplo utiliza la biblioteca de solicitudes de URL de cliente (cURL) para enviar un único documento a un índice denominado `movies-index` dentro de una colección:

```
curl -XPOST \
    --user "$AWS_ACCESS_KEY_ID":"$AWS_SECRET_ACCESS_KEY" \
    --aws-sigv4 "aws:amz:us-east-1:aoss" \
```

```
    --header "x-amz-content-sha256: $REQUEST_PAYLOAD_SHA_HASH" \
    --header "x-amz-security-token: $AWS_SESSION_TOKEN" \
    "https://my-collection-endpoint.us-east-1.aoss.amazonaws.com/movies-index/_doc" \
    -H "Content-Type: application/json" -d '{"title": "Shawshank Redemption"}'
```

Indexación con Postman

La siguiente imagen muestra cómo enviar solicitudes a una colección con Postman. Para obtener instrucciones sobre cómo autenticarse, consulte Authenticate with AWS Signature authentication workflow in Postman.



Configurar Machine Learning en Amazon OpenSearch sin servidor

Machine Learning

Machine Learning (ML) proporciona capacidades de ML en forma de algoritmos de ML y modelos remotos. Con el acceso a estos modelos, puede ejecutar varios flujos de trabajo de IA, como RAG

o la búsqueda semántica. ML permite la experimentación y la implementación en producción de casos de uso de IA generativa mediante el uso de los últimos modelos alojados externamente que se pueden configurar con conectores. Después de configurar un conector, debe configurarlo en un modelo y, a continuación, implementarlo para realizar una predicción.

Connectors

Los conectores facilitan el acceso a los modelos de alojados en plataformas de ML de terceros. Sirven como puerta de enlace entre el clúster de OpenSearch y un modelo remoto. Para obtener más información, consulte la documentación siguiente:

- Creación de conectores para plataformas de ML de terceros en el sitio web de Documentación de OpenSearch
- Conectores para plataformas externas
- Conectores para Servicios de AWS

> **Important**
>
> - Al crear una política de confianza, agregue **`ml.opensearchservice.amazonaws.com`** como principio de OpenSearch Service.
> - Omita los pasos de la página Conectores que muestran cómo configurar un dominio en la política.
> - Agregue la sentencia `iam:PassRole` en el paso de Configuración de permisos.
> - Omita el paso Asignar el rol de ML en OpenSearch Dashboards. No es necesaria la configuración del rol de backend. Esto se aplica a los Conectores para Servicios de AWS y a los Conectores para plataformas externas.
> - En su solicitud de SigV4 al punto de conexión de la colección, defina el nombre del servicio como **`aoss`** en lugar de **`es`**.

Modelos

Un modelo es la funcionalidad principal que se utiliza en varios flujos de trabajo de IA. Por lo general, se asocia el conector a un modelo para realizar una predicción mediante el conector. Una vez que el modelo está en estado desplegando, puede ejecutar la predicción. Para obtener más información, consulte Registrar un modelo alojado en una plataforma de terceros en el sitio web de la Documentación de OpenSearch.

> **Note**
>
> OpenSearch sin servidor no admite todas las características de los modelos, como los modelos locales. Para obtener más información, consulte Características y API de Machine Learning no compatibles.

Configurar permisos para Machine Learning

En la siguiente sección, se describen las políticas de acceso a los datos de recopilación necesarias para Machine Learning (ML). Reemplace los *valores de marcador de posición* con su información específica. Para obtener más información, consulte Permisos de política compatibles.

```
{
    "Rules": [
        {
            "Resource": [
                "model/collection_name/*"
            ],
            "Permission": [
                "aoss:DescribeMLResource",
                "aoss:CreateMLResource",
                "aoss:UpdateMLResource",
                "aoss:DeleteMLResource",
                "aoss:ExecuteMLResource"
            ],
            "ResourceType": "model"
        }
    ],
    "Principal": [
        "arn:aws:iam::account_id:role/role_name"
    ],
    "Description": "ML full access policy for collection_name"
}
```

- aoss:DescribeMLResource: otorga permiso para buscar y consultar conectores, modelos y grupos de modelos.
- aoss:CreateMLResource: otorga permiso para crear conectores, modelos y grupos de modelos.

- aoss:UpdateMLResource: otorga permiso para actualizar conectores, modelos y grupos de modelos.
- aoss:DeleteMLResource: otorga permiso para eliminar conectores, modelos y grupos de modelos.
- aoss:ExecuteMLResource: otorga permiso para realizar predicciones en los modelos.

Características y API de Machine Learning no compatibles

API no admitidas

Amazon OpenSearch sin servidor no admite las siguientes API de Machine Learning (ML):

- Funcionalidad de modelo local
- API de entrenamiento de modelos
- API de algoritmos de predicción de modelos
- API de predicción por lotes
- API de agentes y sus herramientas correspondientes
- API del servidor MCP
- API de memoria
- API de controlador
- API de ejecución de algoritmos
- API de perfil de ML
- API de estadísticas de ML

Para obtener más información sobre las API de ML, consulte la sección API de ML en el sitio web de la documentación de OpenSearch.

Características no admitidas

Las siguientes características de ML no son compatibles con Amazon OpenSearch sin servidor:

- Agentes y herramientas
- Modelos locales
- El procesador de inferencia de machine learning dentro de las canalizaciones de búsqueda y de ingesta
  - Procesador de ingesta de inferencia de ML

- Procesador de respuesta de búsqueda de inferencia de ML
- Procesador de petición de búsqueda de inferencia de ML

Para obtener más información sobre estas características, consulte la siguiente documentación en el sitio web de documentación de OpenSearch.

- Machine learning
- Procesador de inferencias de ML
- Buscar canalizaciones

Configure la búsqueda neuronal y la búsqueda híbrida en OpenSearch sin servidor

Búsqueda neuronal

Amazon OpenSearch sin servidor es compatible con la funcionalidad de búsqueda neuronal para operaciones de búsqueda semántica en sus datos. La búsqueda neuronal utiliza modelos de machine learning para comprender el significado semántico y el contexto de sus peticiones, lo que proporciona resultados de búsqueda más relevantes que las búsquedas tradicionales basadas en palabras clave. Esta sección explica cómo configurar la búsqueda neuronal en OpenSearch sin servidor, incluidas los permisos necesarios, los procesadores compatibles y las diferencias clave con respecto a la implementación estándar de OpenSearch.

Con la búsqueda neuronal, usted puede realizar búsquedas semánticas en sus datos, lo que considera el significado semántico para comprender la intención de sus peticiones de búsqueda. Esta capacidad se basa en los siguientes componentes:

- Procesador de canalización de ingesta de incrustaciones de texto
- Consulta neuronal
- Consulta neuronal dispersa

Búsqueda híbrida

Con la búsqueda híbrida, usted puede mejorar la relevancia de la búsqueda combinando las capacidades de búsqueda por palabras clave y búsqueda semántica. Para usar la búsqueda híbrida,

cree una canalización de búsqueda que procese sus resultados y combine las puntuaciones de los documentos. Para obtener más información, consulte Canalizaciones de búsqueda en el sitio web de la documentación de OpenSearch. Use los siguientes componentes para implementar la búsqueda híbrida:

- Procesador de canalización de búsqueda de normalización

  Técnicas de normalización compatibles

  - `min_max`
  - `l2`

  Técnicas de combinación compatibles

  - `arithmetic_mean`
  - `geometric_mean`
  - `harmonic_mean`

  Para obtener más información sobre las técnicas de normalización y combinación, consulte Campos del cuerpo de la petición en el sitio web de la documentación de OpenSearch.
- Consulta híbrida

Consultas neuronales e híbridas

De forma predeterminada, OpenSearch calcula las puntuaciones de los documentos utilizando el algoritmo Okapi BM25 basado en palabras clave, que funciona bien para peticiones de búsqueda que contienen palabras clave. La búsqueda neuronal proporciona nuevos tipos de consultas para consultas en lenguaje natural y la capacidad de combinar búsqueda semántica y por palabras clave.

Example : **`neural`**

```
"neural": {
  "vector_field": {
    "query_text": "query_text",
    "query_image": "image_binary",
    "model_id": "model_id",
    "k": 100
  }
}
```

Para obtener más información, consulte Consulta neuronal en el sitio web de la documentación de OpenSearch.

Example : **`hybrid`**

```
"hybrid": {
      "queries": [
        array of lexical, neural, or combined queries
      ]
    }
```

Para obtener más información, consulte Consulta híbrida en el sitio web de la documentación de OpenSearch.

Para configurar los componentes de búsqueda semántica en Amazon OpenSearch sin servidor, siga los pasos del tutorial de búsqueda neuronal en el sitio web de la documentación de OpenSearch. Tenga en cuenta estas diferencias importantes:

- OpenSearch sin servidor solo admite modelos remotos. Debe configurar los conectores para los modelos alojados de forma remota. No es necesario implementar o eliminar modelos remotos. Para obtener más información, consulte Introducción a la búsqueda semántica e híbrida en el sitio web de la documentación de OpenSearch.
- Espere hasta 15 segundos de latencia cuando realice búsquedas en su índice vectorial o busque canalizaciones de búsqueda e ingesta creadas recientemente.

Configuración de permisos

La búsqueda neuronal en OpenSearch sin servidor requiere los siguientes permisos. Para obtener más información, consulte Permisos de política compatibles.

Example : Política de búsqueda neuronal

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "NeuralSearch",
            "Effect": "Allow",
            "Action": [
              "aoss:CreateIndex",
              "aoss:CreateCollection",
              "aoss:UpdateCollection",
              "aoss:DeleteIndex",
              "aoss:DeleteCollection"
            ],
            "Resource": "arn:aws:aoss:us-east-1:111122223333:collection/your-collection-name"
        }
    ]
}
```

- aoss:*Index: crea un índice vectorial donde se almacenan las incrustaciones de texto.
- aoss:*CollectionItems: crea y registra modelos de incrustación de texto.
- aoss:*MLResource: crea y registra modelos de incrustación de texto.
- aoss:APIAccessAll: proporciona acceso a las API de OpenSearch para operaciones de búsqueda e ingesta.

A continuación, se describen las políticas de acceso a datos de la colección necesarias para la búsqueda neuronal. Reemplace los *valores de marcador de posición* con su información específica.

```
            {
                "ResourceType": "index",
                "Resource": ["index/collection_name/*"],
                "Permission": [
                  "aoss:CreateIndex",
                  "aoss:DescribeIndex",
                  "aoss:UpdateIndex",
                  "aoss:DeleteIndex"
                ]
            }
        ],
        "Principal": [
            "arn:aws:iam::account_id:role/role_name"
        ]
    },
    {
        "Description": "Create pipeline permission",
        "Rules": [
            {
                "ResourceType": "collection",
                "Resource": ["collection/collection_name"],
                "Permission": [
                  "aoss:CreateCollectionItems",
                  "aoss:DescribeCollectionItems",
                  "aoss:UpdateCollectionItems",
                  "aoss:DeleteCollectionItems"
                ]
            }
        ],
        "Principal": [
            "arn:aws:iam::account_id:role/role_name"
        ]
    },
    {
        "Description": "Create model permission",
        "Rules": [
            {
                "ResourceType": "model",
                "Resource": ["model/collection_name/*"],
                "Permission": ["aoss:CreateMLResources"]
            }
        ],
        "Principal": [
            "arn:aws:iam::account_id:role/role_name"
        ]
    }
]
```

Configure los flujos de trabajo en Amazon OpenSearch sin servidor

Flujos de trabajo

Los flujos de trabajo ayudan a los desarrolladores a innovar con aplicaciones de IA en OpenSearch. El proceso actual de usar las ofertas de machine learning (ML) en OpenSearch, como la búsqueda semántica, requiere tareas de configuración y preprocesamiento complejas, junto con peticiones de usuario extensas, lo cual puede llevar tiempo y ser propenso a errores. Los flujos de trabajo son un marco de simplificación para encadenar múltiples llamadas a las API de OpenSearch.

Para obtener información sobre la configuración y el uso, consulte Automatización de configuraciones en el sitio web de OpenSearch. Cuando usted usa flujos de trabajo en OpenSearch sin servidor, tenga en cuenta estas diferencias importantes:

- OpenSearch sin servidor utiliza únicamente modelos remotos en los pasos del flujo de trabajo. No es necesario implementar estos modelos.
- OpenSearch sin servidor no es compatible con el paso de flujo de trabajo de re indexación.
- Cuando busque flujos de trabajo y estados de flujos de trabajo después de otras llamadas a la API, espere hasta 15 segundos de latencia para que aparezcan las actualizaciones.

Las colecciones de OpenSearch sin servidor son compatibles con los flujos de trabajo cuando estos se utilizan como un origen de datos en su aplicación de la interfaz de usuario de OpenSearch. Para obtener más información, consulte Administración de asociaciones de orígenes de datos.

Configuración de permisos

Antes de crear y aprovisionar una plantilla, verifique que tenga los permisos necesarios. Si necesita ayuda, comuníquese con el administrador de su cuenta. Los flujos de trabajo de OpenSearch sin servidor requieren los siguientes permisos. Usted puede delimitar los permisos a una colección específica definiendo el ARN del recurso de la colección en su política de IAM.

Example : Política de flujos de trabajo

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Sid": "NeuralSearch",
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "arn:aws:iam::111122223333:role/Cognito_identitypoolname/Auth_Role"
        ]
      },
      "Action": [
        "aoss:CreateIndex",
        "aoss:CreateCollection",
        "aoss:UpdateCollection",
        "aoss:DeleteIndex",
        "aoss:DeleteCollection"
      ],
      "Resource": "arn:aws:aoss:us-east-1:111122223333:collection/your-
collection-name"
    }
  ]
}
```

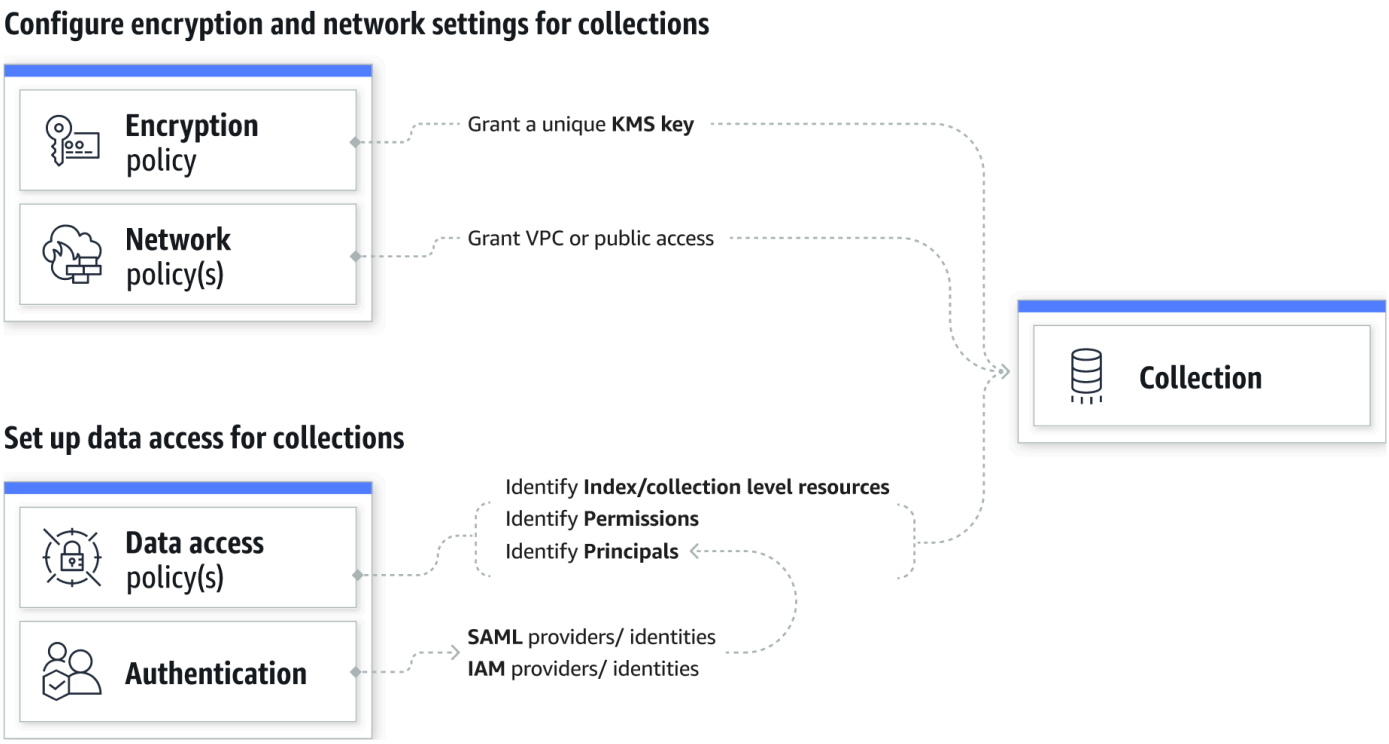
- aoss:*CollectionItems: otorga permiso para crear y administrar plantillas y aprovisionar canalizaciones de búsqueda e ingesta.
- aoss:*Index: otorga permiso para crear y eliminar índices mediante operaciones de la API de OpenSearch.
- aoss:*MLResource: otorga permiso para aprovisionar pasos de flujo de trabajo que utilizan la configuración de machine learning.

Información general sobre la seguridad de Amazon OpenSearch sin servidor

La seguridad de Amazon OpenSearch sin servidor difiere fundamentalmente de la seguridad de Amazon OpenSearch Service en los siguientes aspectos:

Característica	OpenSearch Service	OpenSearch sin servidor
Control de acceso a los datos	El acceso a los datos está determinado por las políticas de IAM y el control de acceso detallado.	El acceso a los datos está determinado por las políticas de acceso a los datos.
Cifrado en reposo	El cifrado en reposo es opcional para los dominios.	El cifrado en reposo es obligator io para las colecciones.
Configuración y administración de seguridad	Debe configurar la red, el cifrado y el acceso a los datos de forma individual para cada dominio.	Puede usar políticas de seguridad para administrar la configuración de seguridad de varias colecciones a escala.

El siguiente diagrama ejemplifica los componentes de seguridad que componen una colección funcional. A una colección se le debe asignar una clave de cifrado, una configuración de acceso a la red y una política de acceso a los datos coincidente que otorgue permisos a sus recursos.



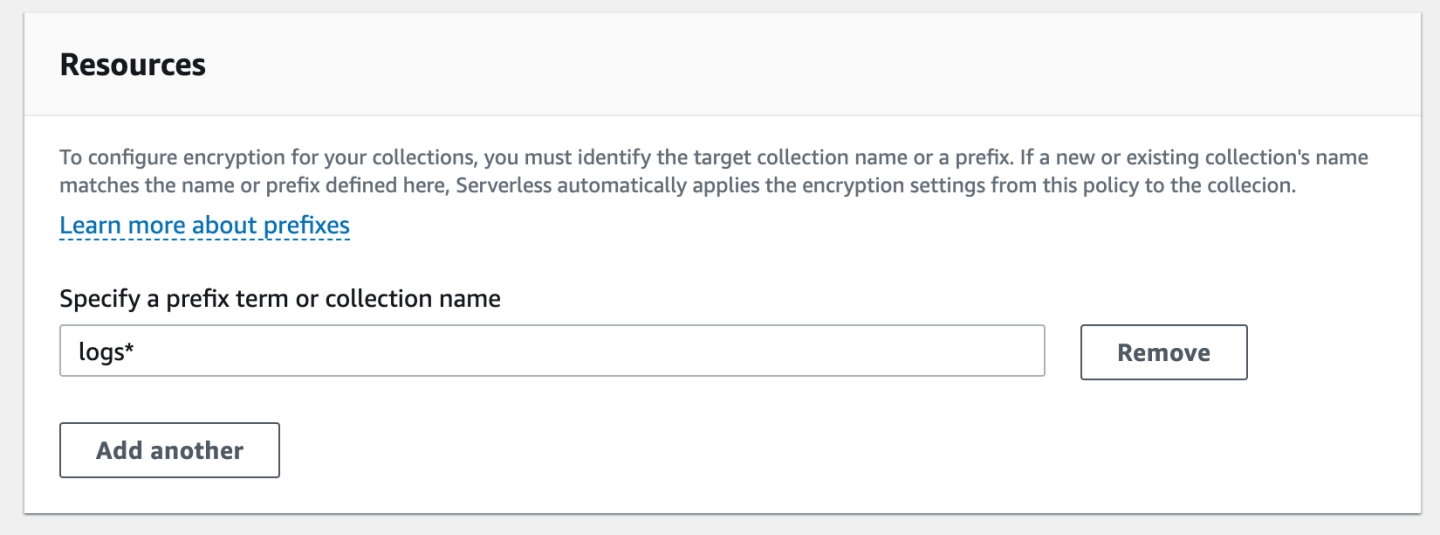
Temas

- Políticas de cifrado
- Políticas de red
- Políticas de acceso a datos
- Autenticación SAML e IAM
- Seguridad de la infraestructura
- Introducción a Amazon OpenSearch sin servidor
- Identity and Access Management en Amazon OpenSearch sin servidor
- Cifrado en Amazon OpenSearch sin servidor
- Acceso a la red para Amazon OpenSearch sin servidor
- Cumplimiento FIPS en Amazon OpenSearch sin servidor
- Control de acceso a los datos para Amazon OpenSearch sin servidor
- Acceda a Amazon OpenSearch sin servidor mediante un punto de conexión de la interfaz (AWS PrivateLink)
- Autenticación SAML para Amazon OpenSearch sin servidor
- Validación de conformidad para Amazon OpenSearch sin servidor

Políticas de cifrado

Las políticas de cifrado definen si las colecciones se cifran con una clave Clave propiedad de AWS o con una clave gestionada por el cliente. Las políticas de cifrado constan de dos componentes: un patrón de recursos y una clave de cifrado. El patrón de recursos define a qué colección o colecciones se aplica la política. La clave de cifrado determina cómo se protegerán las colecciones asociadas.

Para aplicar una política a varias colecciones debe incluir un comodín (*) en la regla de política. Por ejemplo, la siguiente política se aplica a todas las colecciones cuyos nombres comiencen por “logs” (registros).

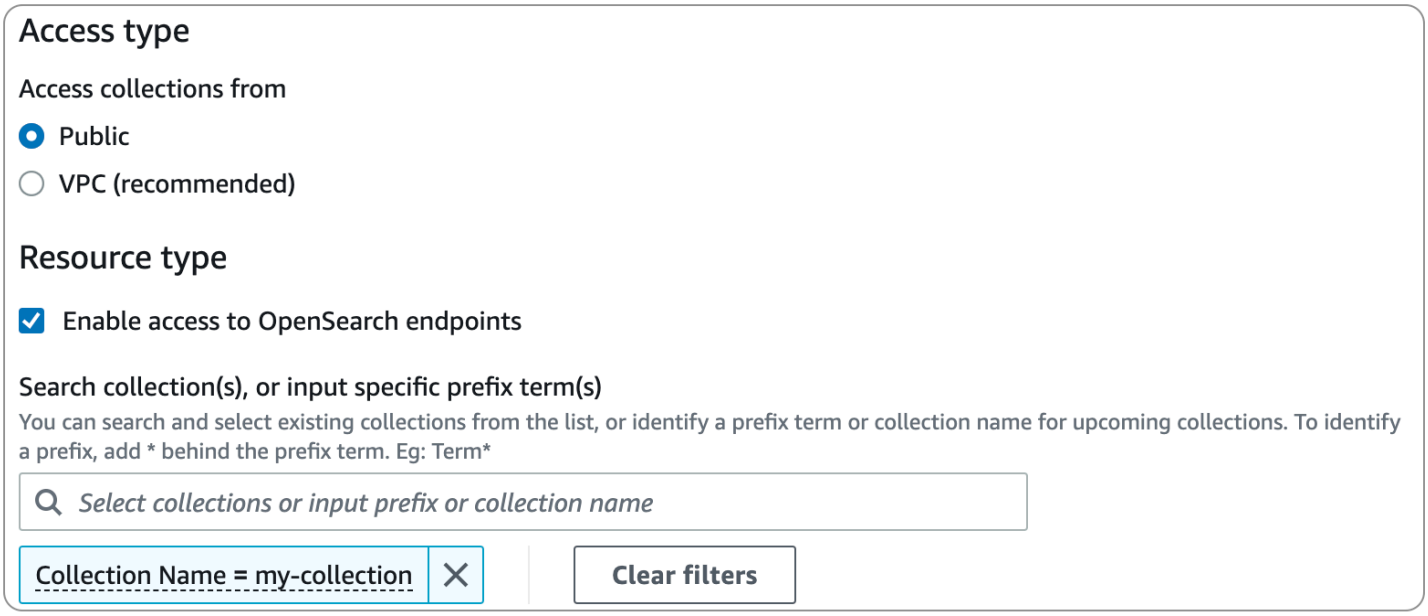


Las políticas de cifrado agilizan el proceso de creación y administración de colecciones, especialmente cuando se hace mediante programación. Puede crear una colección especificando un nombre y, al crearla, se le asigna de forma automática una clave de cifrado.

Políticas de red

Las políticas de red definen si se puede acceder a las colecciones de forma privada o a través de Internet desde redes públicas. Se puede acceder a colecciones privadas a través de puntos de conexión de VPC administrados por OpenSearch sin servidor o mediante Servicios de AWS específicos, como Amazon Bedrock con acceso privado al Servicio de AWS. Al igual que las políticas de cifrado, las políticas de red se pueden aplicar a varias colecciones, lo que le permite administrar el acceso a la red para muchas colecciones a gran escala.

Las políticas de red constan de dos componentes: un tipo de acceso y un tipo de recurso. El tipo de acceso puede ser público o privado. El tipo de recurso determina si el acceso que elija se aplica al punto de conexión de la colección, al punto de conexión de OpenSearch Dashboards o ambos.



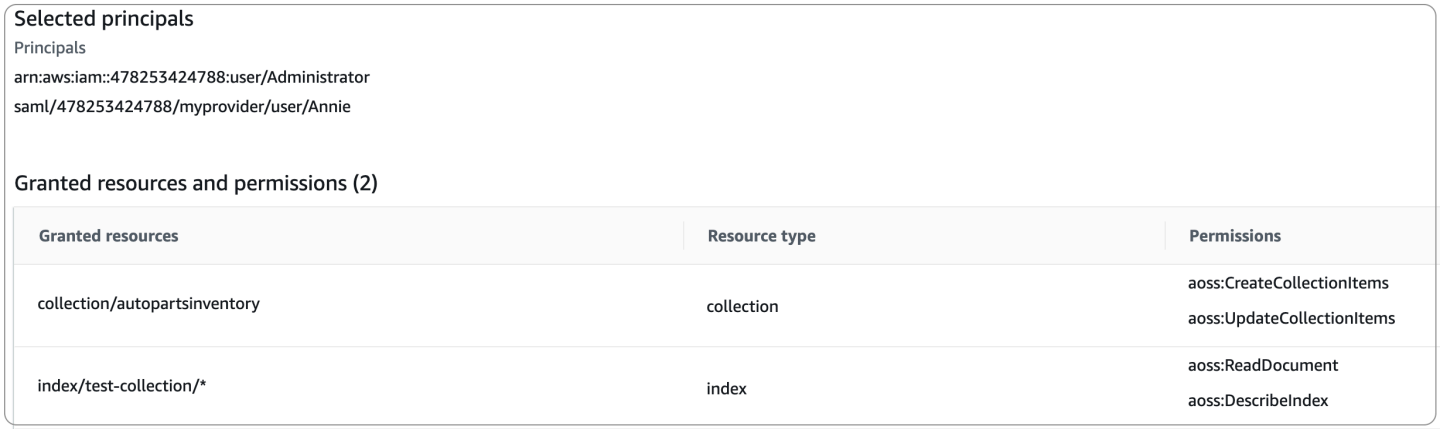
Si planea configurar el acceso de VPC dentro de una política de red, primero debe crear uno o más puntos de conexión de VPC administrados por OpenSearch sin servidor. Estos puntos de conexión le permiten acceder a OpenSearch sin servidor como si estuviera en su VPC, sin necesidad de utilizar una puerta de enlace de Internet, un dispositivo NAT, una conexión VPN o una conexión Direct Connect.

El acceso privado a los Servicios de AWS se aplica solo al punto de conexión de OpenSearch de la colección, no al punto de conexión de OpenSearch Dashboards. No se puede conceder acceso a Servicios de AWS a OpenSearch Dashboards.

Políticas de acceso a datos

Las políticas de acceso a datos definen la forma en que los usuarios acceden a los datos de sus colecciones. Las políticas de acceso a datos le ayudan a administrar las colecciones a escala mediante la asignación automática de permisos de acceso a las colecciones e índices que coinciden con un patrón específico. Se pueden aplicar varias políticas a un solo recurso.

Las políticas de acceso a datos constan de un conjunto de reglas, cada una con tres componentes: un tipo de recurso, los recursos concedidos y un conjunto de permisos. El tipo de recurso puede ser una colección o un índice. Los recursos concedidos pueden ser nombres o patrones de colecciones o índices con un comodín (*). La lista de permisos especifica a qué operaciones de la API de OpenSearch le concede acceso esta política. Además, la política contiene una lista de entidades principales, que especifican los roles, los usuarios y las identidades SAML de IAM a los que se debe conceder acceso.



Para obtener más información sobre el formato de una política de acceso a datos, consulte la sintaxis de la política.

Antes de crear una política de acceso a datos, debe tener uno o más roles o usuarios de IAM, o identidades SAML, a los que proporcionar acceso en la política. Para más información, consulte la siguiente sección.

> **Note**
>
> Cambiar de Acceso Público a Acceso Privado para su colección eliminará la pestaña Índices en la Consola de Colecciones de OpenSearch sin servidor.

Autenticación SAML e IAM

La entidad principal de IAM y las identidades de SAML son uno de los componentes básicos de una política de acceso a los datos. En la instrucción `principal` de una política de acceso puede incluir roles, usuarios e identidades SAML de IAM. A estas entidades principales se le conceden los permisos que usted especifique en las reglas de política asociadas.

```
[
   {
      "Rules":[
         {
            "ResourceType":"index",
            "Resource":[
               "index/marketing/orders*"
            ],
            "Permission":[
```

```
                "aoss:*"
            ]
         }
      ],
      "Principal":[
         "arn:aws:iam::123456789012:user/Dale",
         "arn:aws:iam::123456789012:role/RegulatoryCompliance",
         "saml/123456789012/myprovider/user/Annie"
      ]
   }
]
```

La autenticación SAML se configura directamente en OpenSearch sin servidor. Para obtener más información, consulte the section called “Autenticación SAML”.

Seguridad de la infraestructura

Amazon OpenSearch sin servidor está protegido por la seguridad de red global de AWS. Para obtener información sobre los servicios de seguridad de AWS y sobre cómo AWS protege la infraestructura, consulte Seguridad en la nube de AWS. Para diseñar su entorno de AWS siguiendo las prácticas recomendadas de seguridad de infraestructura, consulte Protección de la infraestructura en Portal de seguridad de AWS Well-Architected Framework.

Puede utilizar llamadas a la API publicadas de AWS de Amazon OpenSearch sin servidor a través de la red. Los clientes deben admitir Transport Layer Security (TLS). Se recomienda el uso de TLS 1.2 y recomendamos TLS 1.3. Para obtener una lista de los cifrados compatibles con TLS 1.3, consulte los protocolos y cifrados TLS en la documentación de Elastic Load Balancing.

Además, debe firmar las solicitudes mediante un ID de clave de acceso y una clave de acceso secreta que esté asociada a una entidad principal de IAM. También puede utilizar AWS Security Token Service (AWS STS) para generar credenciales de seguridad temporales para firmar solicitudes.

Introducción a Amazon OpenSearch sin servidor

Los siguientes tutoriales lo ayudan a empezar a utilizar Amazon OpenSearch sin servidor. Ambos tutoriales siguen los mismos pasos básicos, pero uno usa la consola mientras que el otro usa la AWS CLI.

Tenga en cuenta que los casos de uso de estos tutoriales están simplificados. Las políticas de red y seguridad son bastante abiertas. En las cargas de trabajo de producción, se recomienda configurar

funciones de seguridad más sólidas, como la autenticación SAML, el acceso a la VPC y las políticas de acceso restrictivo a los datos.

Temas

- Tutorial: Introducción a la seguridad en Amazon OpenSearch sin servidor (consola)
- Tutorial: Introducción a la seguridad en Amazon OpenSearch sin servidor (CLI)

Tutorial: Introducción a la seguridad en Amazon OpenSearch sin servidor (consola)

En este tutorial, se muestran los pasos básicos para crear y administrar políticas de seguridad con la consola de Amazon OpenSearch sin servidor.

En este tutorial, debe completar los siguientes pasos:

1. Configurar permisos
2. Crear una política de cifrado
3. Crear una política de red
4. Configurar la política de acceso a datos
5. Crear una colección
6. Cargar y buscar datos

Este tutorial le mostrará cómo configurar una colección con Consola de administración de AWS. Para conocer los mismos pasos que se siguen al utilizar AWS CLI, consulte the section called “Tutorial: Introducción a la seguridad (CLI)”.

Paso 1: Configurar los permisos

> **Note**
>
> Puedes omitir este paso si ya utilizas una política más amplia basada en la identidad, como `Action":"aoss:*"` o `Action":"*"`. Sin embargo, en entornos de producción, le recomendamos que siga la entidad principal del privilegio mínimo y solo asigne los permisos mínimos necesarios para completar una tarea.

Para completar este tutorial, debe tener los permisos de IAM correctos. Su usuario o rol debe tener adjunta una política basada en la identidad con los siguientes permisos mínimos:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Action": [
        "aoss:ListCollections",
        "aoss:BatchGetCollection",
        "aoss:CreateCollection",
        "aoss:CreateSecurityPolicy",
        "aoss:GetSecurityPolicy",
        "aoss:ListSecurityPolicies",
        "aoss:CreateAccessPolicy",
        "aoss:GetAccessPolicy",
        "aoss:ListAccessPolicies"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

Para obtener una lista completa de los permisos de OpenSearch sin servidor, consulte the section called “Identity and Access Management”.

Paso 2: Crear una política de cifrado

Las políticas de cifrado especifican la clave AWS KMS que utilizará OpenSearch sin servidor para cifrar la colección. Puede cifrar colecciones con una clave Clave administrada de AWS o una clave diferente. Para simplificar este tutorial, cifraremos nuestra colección con un Clave administrada de AWS.

Para crear una política de cifrado

1. Abra la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. Expanda Sin servidor en el panel de navegación de la izquierda y seleccione Políticas de red.
3. Seleccione Crear política de cifrado.

4. Asigne el nombre de esta política a los libros de políticas. Para la descripción, introduzca Política de cifrado para la colección de libros.
5. En Recursos, introduzca libros, que es el nombre con el que llamará su colección. Si desea que este sea más extenso, puede incluir un asterisco (`books*`) para que la política se aplique a todas las colecciones que comiencen por la palabra “libros”.
6. En Cifrado, mantenga seleccionado Utilizar clave propia de AWS.
7. Seleccione Crear.

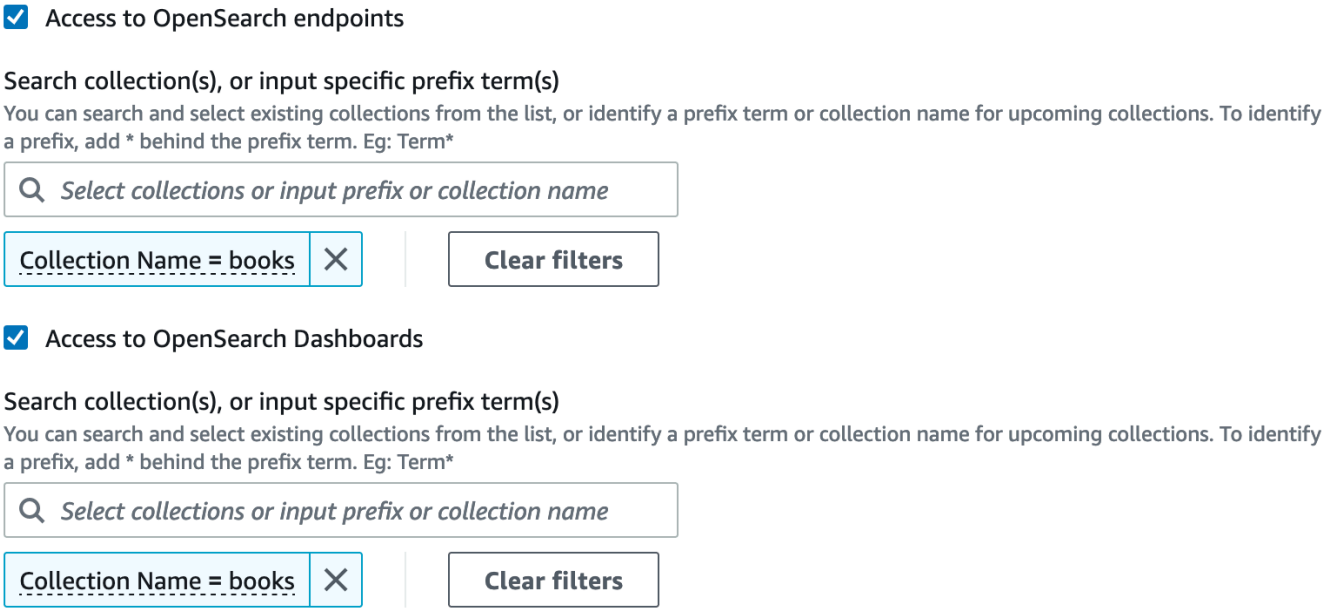
Paso 3: crear una política de red

Las políticas de red determinan si se puede acceder a las colecciones a través de Internet desde redes públicas, o si debe acceder a ella a través de puntos de conexión de VPC gestionados por OpenSearch sin servidor. En este tutorial, configuraremos el acceso público.

Crear una política de red

1. Seleccione Políticas en el panel de navegación de la izquierda y, a continuación, Crear política.
2. Asigne el nombre de esta política a los libros de políticas. Para la descripción, introduzca Política de redes para la colección de libros.
3. En la Regla 1, asígnele un nombre a la regla Acceso público a la colección de libros.
4. Para simplificar este tutorial, configuraremos el acceso público para la colección de libros. Para el tipo de acceso, seleccione Público.
5. Vamos a acceder a la colección desde OpenSearch Dashboards. Para ello, debe configurar el acceso a la red de Dashboards y el punto de conexión de OpenSearch; de lo contrario, Dashboards no funcionará.

   Para el tipo de recurso, habilite tanto el acceso a los puntos de conexión de OpenSearch como el acceso a los cuadros de mando de OpenSearch.
6. En ambos cuadros de entrada, introduzca Nombre de la colección = libros. Esta configuración reduce el alcance de la política para que solo se aplique a una única colección (`books`). La regla debe tener un aspecto similar al siguiente:



7. Seleccione Crear.

Paso 4: crear una política de acceso a datos

No podrá acceder a los datos de su colección hasta que configure el acceso a los datos. Las políticas de acceso a los datos son independientes de la política basada en identidades de IAM que configuró en el paso 1. Permiten a los usuarios acceder a los datos reales de una colección.

En este tutorial, le brindamos a un solo usuario los permisos necesarios para indexar datos en la colección de libros.

Para crear una política de acceso a datos

1. En el panel de navegación de la izquierda, seleccione Políticas de acceso a datos y Crear política de acceso.
2. Asigne el nombre de esta política a los libros de políticas. Para la descripción, introduzca Política de acceso de datos para la colección de libros.
3. Seleccione JSON para el método de definición de políticas y pegue la siguiente política en el editor de JSON.

   Sustituya la entidad principal de ARN por el ARN de la cuenta que utilizará para iniciar sesión en OpenSearch Dashboards e indexar los datos.

```
[
```

```
   {
      "Rules":[
         {
            "ResourceType":"index",
            "Resource":[
               "index/books/*"
            ],
            "Permission":[
               "aoss:CreateIndex",
               "aoss:DescribeIndex",
               "aoss:ReadDocument",
               "aoss:WriteDocument",
               "aoss:UpdateIndex",
               "aoss:DeleteIndex"
            ]
         }
      ],
      "Principal":[
         "arn:aws:iam::123456789012:user/my-user"
      ]
   }
]
```

Esta política proporciona a un solo usuario los permisos mínimos necesarios para crear un índice en la colección de libros, indexar algunos datos y buscarlos.

4. Seleccione Crear.

Paso 5: crear una colección

Como ya configuró las políticas de cifrado y red, puede crear una colección coincidente y la configuración de seguridad se le aplicará automáticamente.

Para crear una colección de OpenSearch sin servidor

1. Seleccione Colecciones en el panel de navegación de la izquierda y elija Crear colección.
2. Nombre los libros de la colección.
3. Para el tipo de colección, seleccione Buscar.
4. En Cifrado, OpenSearch sin servidor le informa de que el nombre de la colección coincide con la política de cifrado de `books-policy`.

5. En Configuración de acceso a la red, OpenSearch sin servidor le informa de que el nombre de la colección coincide con la política de red de `books-policy`.
6. Seleccione Siguiente.
7. En Opciones de política de acceso a datos, OpenSearch sin servidor informa de que el nombre de la colección coincide con la política de acceso a datos `books-policy`.
8. Seleccione Siguiente.
9. Revise la configuración de la colección y seleccione Enviar. Las colecciones suelen tardar menos de un minuto en inicializarse.

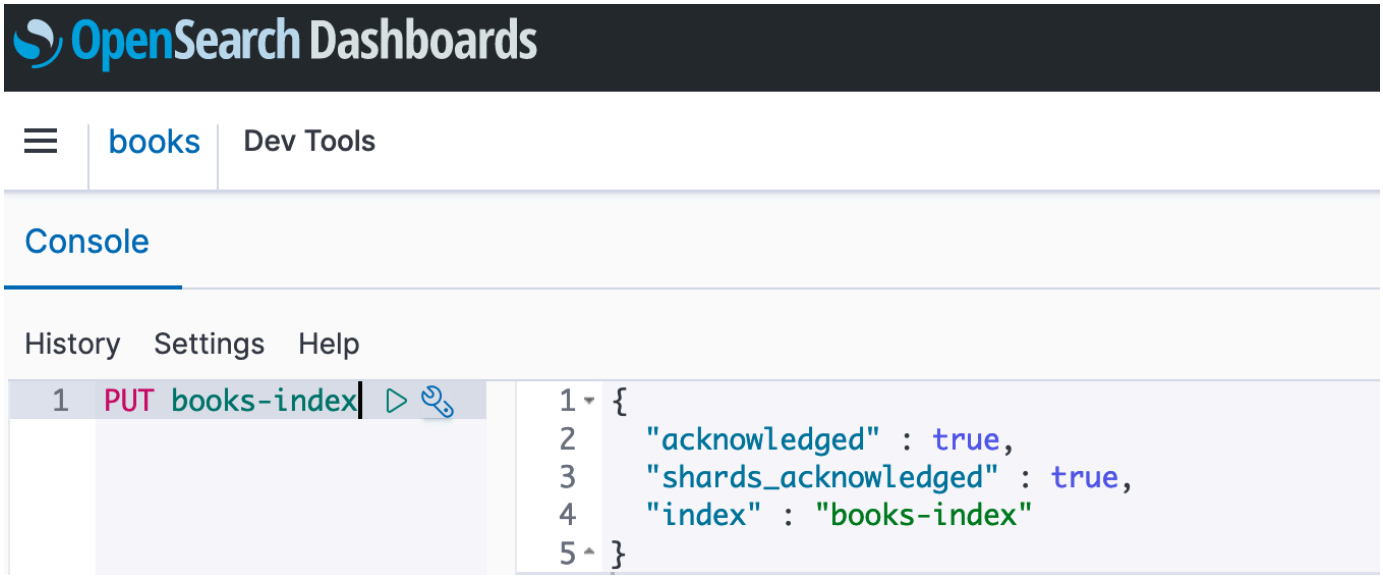
Paso 6: Cargar y buscar datos

Puede cargar datos en una colección de OpenSearch sin servidor con Postman o curl. Para ser breves, en estos ejemplos se utilizan las herramientas de desarrollo en la consola de OpenSearch Dashboards.

Para indexar y buscar datos en la colección

1. Seleccione Colecciones en el panel de navegación de la izquierda y elija la colección de libros para abrir su página de detalles.
2. Seleccione la URL de OpenSearch Dashboards para la colección. La URL toma el formato `https://collection-id.us-east-1.aoss.amazonaws.com/_dashboards`.
3. Inicie sesión con las claves secretas y de acceso AWS de la entidad principal que especificó en su política de acceso a datos.
4. En OpenSearch Dashboards, abra el menú de navegación izquierdo y seleccione Herramientas de desarrollo.
5. Para crear un índice único denominado books-index, ejecute el siguiente comando:

```
PUT books-index
```



6. Para indexar un solo documento en books-index, ejecute el siguiente comando:

```
PUT books-index/_doc/1
{
  "title": "The Shining",
  "author": "Stephen King",
  "year": 1977
}
```

7. Para buscar datos en OpenSearch Dashboards, debe configurar al menos un patrón de índice. OpenSearch utiliza estos patrones para identificar los índices que desea analizar. Abra el menú principal de Dashboards, seleccione Gestión de pilas, luego seleccione Patrones de índice y, a continuación, seleccione Crear patrón de índice. Para este tutorial, introduzca books-index.
8. seleccione Siguiente paso y, a continuación, seleccione Crear patrón de índice. Una vez creado el patrón, puede ver los diversos campos de documento, como `author` y `title`.
9. Para comenzar a buscar los datos, vuelva a abrir el menú principal y elija Explorar o utilice la API de búsqueda.

Tutorial: Introducción a la seguridad en Amazon OpenSearch sin servidor (CLI)

Este tutorial explica los pasos descritos en el tutorial de Introducción a la consola sobre seguridad, pero utiliza AWS CLI en lugar de la consola de OpenSearch Service.

En este tutorial, deberá completar los siguientes pasos:

1. Creación de una política de permisos de IAM
2. Anexo de la política de IAM a un rol de IAM
3. Crear una política de cifrado
4. Crear una política de red
5. Crear una recopilación
6. Configurar la política de acceso a datos
7. Recuperación del punto de conexión de la colección
8. Carga de los datos a su conexión
9. Búsqueda de datos en su colección

El objetivo de este tutorial es configurar una única colección de OpenSearch sin servidor con configuraciones de cifrado, red y acceso a datos bastante simples. Por ejemplo, configuraremos el acceso a la red pública, el cifrado Clave administrada de AWS y una política de acceso a datos simplificada que otorgue permisos mínimos a un solo usuario.

En un escenario de producción, considere implementar una configuración más robusta, incluyendo autenticación SAML, una clave de cifrado personalizada y acceso VPC.

Para empezar con las políticas de seguridad en OpenSearch sin servidor

1.
> **Note**
>
> Puede omitir este paso si ya utilizas una política más amplia basada en la identidad, como `Action":"aoss:*"` o `Action":"*"`. Sin embargo, en entornos de producción, le recomendamos que siga la entidad principal del privilegio mínimo y solo asigne los permisos mínimos necesarios para completar una tarea.

   Para empezar, cree una política AWS Identity and Access Management con los permisos mínimos necesarios para realizar los pasos de este tutorial. Le pondremos a la política el siguiente nombre `TutorialPolicy`:

```
aws iam create-policy \
  --policy-name TutorialPolicy \
  --policy-document "{\"Version\": \"2012-10-17\",\"Statement\":
 [{\"Action\": [\"aoss:ListCollections\",\"aoss:BatchGetCollection\",
```

```
\"aoss:CreateCollection\",\"aoss:CreateSecurityPolicy\",\"aoss:GetSecurityPolicy\",
\"aoss:ListSecurityPolicies\",\"aoss:CreateAccessPolicy\",\"aoss:GetAccessPolicy\",
\"aoss:ListAccessPolicies\"],\"Effect\": \"Allow\",\"Resource\": \"*\"}]}"
```

Respuesta de ejemplo

```
{
    "Policy": {
        "PolicyName": "TutorialPolicy",
        "PolicyId": "ANPAW6WRAECKG6QJWUV7U",
        "Arn": "arn:aws:iam::123456789012:policy/TutorialPolicy",
        "Path": "/",
        "DefaultVersionId": "v1",
        "AttachmentCount": 0,
        "PermissionsBoundaryUsageCount": 0,
        "IsAttachable": true,
        "CreateDate": "2022-10-16T20:57:18+00:00",
        "UpdateDate": "2022-10-16T20:57:18+00:00"
    }
}
```

2. Asocie `TutorialPolicy` al rol de IAM que indexará y buscará datos en la colección. Le pondremos al usuario el siguiente nombre `TutorialRole`:

```
aws iam attach-role-policy \
  --role-name TutorialRole \
  --policy-arn arn:aws:iam::123456789012:policy/TutorialPolicy
```

3. Antes de crear una colección, debe crear una política de cifrado que asigne una Clave propiedad de AWS a la colección de libros que creará en un paso posterior.

   Envíe la siguiente solicitud para crear una política de cifrado para la colección de libros:

```
aws opensearchserverless create-security-policy \
  --name books-policy \
  --type encryption --policy "{\"Rules\":[{\"ResourceType\":\"collection\",
\"Resource\":[\"collection\/books\"]}],\"AWSOwnedKey\":true}"
```

   Respuesta de ejemplo

```
{
```

```
    "securityPolicyDetail": {
        "type": "encryption",
        "name": "books-policy",
        "policyVersion": "MTY2OTI0MDAwNTk5MF8x",
        "policy": {
            "Rules": [
                {
                    "Resource": [
                        "collection/books"
                    ],
                    "ResourceType": "collection"
                }
            ],
            "AWSOwnedKey": true
        },
        "createdDate": 1669240005990,
        "lastModifiedDate": 1669240005990
    }
}
```

4. Cree una política de red que proporcione acceso público a la colección de libros:

```
aws opensearchserverless create-security-policy --name books-policy --type network
 \
  --policy "[{\"Description\":\"Public access for books collection\",\"Rules
\":[{\"ResourceType\":\"dashboard\",\"Resource\":[\"collection\/books\"]},
{\"ResourceType\":\"collection\",\"Resource\":[\"collection\/books\"]}],
\"AllowFromPublic\":true}]"
```

Respuesta de ejemplo

```
{
    "securityPolicyDetail": {
        "type": "network",
        "name": "books-policy",
        "policyVersion": "MTY2OTI0MDI1Njk1NV8x",
        "policy": [
            {
                "Rules": [
                    {
                        "Resource": [
                            "collection/books"
                        ],
```

```
                        "ResourceType": "dashboard"
                    },
                    {
                        "Resource": [
                            "collection/books"
                        ],
                        "ResourceType": "collection"
                    }
                ],
                "AllowFromPublic": true,
                "Description": "Public access for books collection"
            }
        ],
        "createdDate": 1669240256955,
        "lastModifiedDate": 1669240256955
    }
}
```

5. Crea la colección de libros:

```
aws opensearchserverless create-collection --name books --type SEARCH
```

Respuesta de ejemplo

```
{
    "createCollectionDetail": {
        "id": "8kw362bpwg4gx9b2f6e0",
        "name": "books",
        "status": "CREATING",
        "type": "SEARCH",
        "arn": "arn:aws:aoss:us-
east-1:123456789012:collection/8kw362bpwg4gx9b2f6e0",
        "kmsKeyArn": "auto",
        "createdDate": 1669240325037,
        "lastModifiedDate": 1669240325037
    }
}
```

6. Cree una política de acceso a datos que proporcione los permisos mínimos para indexar y buscar datos en la colección de libros. Reemplace entidad principal de ARN por el ARN de `TutorialRole` del paso 1:

```
aws opensearchserverless create-access-policy \
  --name books-policy \
  --type data \
  --policy "[{\"Rules\":[{\"ResourceType\":\"index\",\"Resource\":
[\"index\/books\/books-index\"],\"Permission\":[\"aoss:CreateIndex
\",\"aoss:DescribeIndex\",\"aoss:ReadDocument\",\"aoss:WriteDocument
\",\"aoss:UpdateIndex\",\"aoss:DeleteIndex\"]}],\"Principal\":
[\"arn:aws:iam::123456789012:role\/TutorialRole\"]}]"
```

Respuesta de ejemplo

```
{
    "accessPolicyDetail": {
        "type": "data",
        "name": "books-policy",
        "policyVersion": "MTY2OTI0MDM5NDY1M18x",
        "policy": [
            {
                "Rules": [
                    {
                        "Resource": [
                            "index/books/books-index"
                        ],
                        "Permission": [
                            "aoss:CreateIndex",
                            "aoss:DescribeIndex",
                            "aoss:ReadDocument",
                            "aoss:WriteDocument",
                            "aoss:UpdateDocument",
                            "aoss:DeleteDocument"
                        ],
                        "ResourceType": "index"
                    }
                ],
                "Principal": [
                    "arn:aws:iam::123456789012:role/TutorialRole"
                ]
            }
        ],
        "createdDate": 1669240394653,
        "lastModifiedDate": 1669240394653
    }
```

```
}
```

`TutorialRole` ahora debería poder indexar y buscar documentos en la colección de libros.

7. Para realizar llamadas a la API de OpenSearch, necesita el punto de conexión de la colección. Envíe la siguiente solicitud para recuperar el parámetro `collectionEndpoint`:

```
aws opensearchserverless batch-get-collection --names books
```

Respuesta de ejemplo

```
{
    "collectionDetails": [
        {
            "id": "8kw362bpwg4gx9b2f6e0",
            "name": "books",
            "status": "ACTIVE",
            "type": "SEARCH",
            "description": "",
            "arn": "arn:aws:aoss:us-
east-1:123456789012:collection/8kw362bpwg4gx9b2f6e0",
            "createdDate": 1665765327107,
            "collectionEndpoint": "https://8kw362bpwg4gx9b2f6e0.us-
east-1.aoss.amazonaws.com",
            "dashboardEndpoint": "https://8kw362bpwg4gx9b2f6e0.us-
east-1.aoss.amazonaws.com/_dashboards"
        }
    ],
    "collectionErrorDetails": []
}
```

> **Note**
>
> No podrá ver el punto de conexión de la colección hasta que el estado de la colección cambie a `ACTIVE`. Puede que tenga que hacer varias llamadas para comprobar el estado hasta que la colección se haya creado de forma correcta.

8. Utilice una herramienta HTTP como Postman o curl para indexar los datos de la colección de libros. Crearemos un índice llamado books-index y agregaremos un solo documento.

Envíe la siguiente solicitud al punto de conexión de la colección que recuperó en el paso anterior, con las credenciales `TutorialRole`.

```
PUT https://8kw362bpwg4gx9b2f6e0.us-east-1.aoss.amazonaws.com/books-index/_doc/1
{
  "title": "The Shining",
  "author": "Stephen King",
  "year": 1977
}
```

Respuesta de ejemplo

```
{
  "_index" : "books-index",
  "_id" : "1",
  "_version" : 1,
  "result" : "created",
  "_shards" : {
    "total" : 0,
    "successful" : 0,
    "failed" : 0
  },
  "_seq_no" : 0,
  "_primary_term" : 0
}
```

9. Para empezar a buscar datos en tu colección, utilice la API de búsqueda. La siguiente consulta representa una búsqueda básica:

```
GET https://8kw362bpwg4gx9b2f6e0.us-east-1.aoss.amazonaws.com/books-index/_search
```

Respuesta de ejemplo

```
{
    "took": 405,
    "timed_out": false,
    "_shards": {
        "total": 6,
        "successful": 6,
        "skipped": 0,
```

```
        "failed": 0
    },
    "hits": {
        "total": {
            "value": 2,
            "relation": "eq"
        },
        "max_score": 1.0,
        "hits": [
            {
                "_index": "books-index:0::3xJq14MBUaOS0wL26UU9:0",
                "_id": "F_bt4oMBLle5pYmm5q4T",
                "_score": 1.0,
                "_source": {
                    "title": "The Shining",
                    "author": "Stephen King",
                    "year": 1977
                }
            }
        ]
    }
}
```


Identity and Access Management en Amazon OpenSearch sin servidor

AWS Identity and Access Management (IAM) es un Servicio de AWS que ayuda a los administradores a controlar de forma segura el acceso a los recursos de AWS. Los administradores de IAM controlan quién puede autenticarse (iniciar sesión) y autorizar (tener permisos) para utilizar los recursos de OpenSearch sin servidor. IAM es un servicio de Servicio de AWS que se puede utilizar sin cargo adicional.

Temas

- Políticas basadas en identidades de OpenSearch sin servidor
- Acciones de política de OpenSearch sin servidor
- Recursos de políticas para OpenSearch sin servidor
- Claves de condición de política para Amazon OpenSearch sin servidor
- ABAC con OpenSearch sin servidor
- Uso de credenciales temporales con OpenSearch sin servidor

- Uso de roles vinculados a servicios para OpenSearch sin servidor
- Otros tipos de políticas
- Ejemplos de políticas basadas en identidades para OpenSearch sin servidor
- Compatibilidad de Centro de identidades de IAM con Amazon OpenSearch sin servidor

Políticas basadas en identidades de OpenSearch sin servidor

Compatibilidad con las políticas basadas en identidad: sí

Las políticas basadas en identidad son documentos de políticas de permisos JSON que puede asociar a una identidad, como un usuario de IAM, un grupo de usuarios o un rol. Estas políticas controlan qué acciones pueden realizar los usuarios y los roles, en qué recursos y en qué condiciones. Para obtener más información sobre cómo crear una política basada en identidad, consulte Creación de políticas de IAM en la Guía del usuario de IAM.

Con las políticas basadas en identidades de IAM, puede especificar las acciones y los recursos permitidos o denegados, así como las condiciones en las que se permiten o deniegan las acciones. Para obtener más información sobre los elementos que puede utilizar en una política de JSON, consulte Referencia de los elementos de las políticas de JSON de IAM en la Guía del usuario de IAM.

Ejemplos de políticas basadas en identidades para OpenSearch sin servidor

Para ver ejemplos de políticas basadas en identidad de OpenSearch sin servidor, consulte the section called “Ejemplos de políticas basadas en identidades”.

Acciones de política de OpenSearch sin servidor

Compatibilidad con las acciones de políticas: sí

El elemento `Action` de una política JSON describe las acciones que puede utilizar para conceder o denegar el acceso en una política. Las acciones de la política generalmente tienen el mismo nombre que la operación de API de AWS asociada. Hay algunas excepciones, como acciones de solo permiso que no tienen una operación de API coincidente. También hay algunas operaciones que requieren varias acciones en una política. Estas acciones adicionales se denominan acciones dependientes.

Incluya acciones en una política para conceder permisos y así llevar a cabo la operación asociada.

Las acciones de políticas de OpenSearch sin servidor utilizan el siguiente prefijo antes de la acción:

```
aoss
```

Para especificar varias acciones en una única instrucción, sepárelas con comas.

```
"Action": [
      "aoss:action1",
      "aoss:action2"
         ]
```

Puede especificar varias acciones utilizando caracteres comodín (*). Por ejemplo, para especificar todas las acciones que comiencen con la palabra `Describe`, incluya la siguiente acción:

```
"Action": "aoss:List*"
```

Para ver ejemplos de políticas basadas en identidad de OpenSearch sin servidor, consulte Ejemplos de políticas basadas en identidades para OpenSearch sin servidor.

Recursos de políticas para OpenSearch sin servidor

Compatibilidad con los recursos de políticas: sí

Los administradores pueden utilizar las políticas JSON de AWS para especificar quién tiene acceso a qué. Es decir, qué entidad principal puede realizar acciones en qué recursos y en qué condiciones.

El elemento `Resource` de la política JSON especifica el objeto u objetos a los que se aplica la acción. Como práctica recomendada, especifique un recurso utilizando el Nombre de recurso de Amazon (ARN). Para las acciones que no admiten permisos por recurso, utilice un carácter comodín (*) para indicar que la instrucción se aplica a todos los recursos.

```
"Resource": "*"
```

Claves de condición de política para Amazon OpenSearch sin servidor

Compatibilidad con claves de condición de políticas específicas del servicio: sí

Los administradores pueden utilizar las políticas JSON de AWS para especificar quién tiene acceso a qué. Es decir, qué entidad principal puede realizar acciones en qué recursos y en qué condiciones.

El elemento `Condition` especifica cuándo se ejecutan las instrucciones en función de criterios definidos. Puede crear expresiones condicionales que utilizan operadores de condición, tales como

igual o menor que, para que la condición de la política coincida con los valores de la solicitud. Para ver todas las claves de condición globales de AWS, consulte Claves de contexto de condición globales de AWS en la Guía del usuario de IAM.

Además del control de acceso basado en atributos (ABAC), OpenSearch sin servidor admite las siguientes claves de condición:

- `aoss:collection`
- `aoss:CollectionId`
- `aoss:index`

Puede utilizar estas claves de condición incluso al proporcionar políticas de acceso y de seguridad. Por ejemplo:

```
[
   {
      "Effect":"Allow",
      "Action":[
         "aoss:CreateAccessPolicy",
         "aoss:CreateSecurityPolicy"
      ],
      "Resource":"*",
      "Condition":{
         "StringLike":{
            "aoss:collection":"log"
         }
      }
   }
]
```

En este ejemplo, la condición se aplica a las políticas que contienen reglas que coinciden tanto con el nombre como con el patrón de una colección. Las condiciones funcionan de la el siguiente manera:

- `StringEquals`: se aplica a políticas con reglas que contienen la cadena de exacta de recursos “log” (es decir,`collection/log`).
- `StringLike`: se aplica a políticas con reglas que contienen la cadena de recursos que incluye la cadena “log” (es decir, `collection/log` pero también `collection/logs-application` o `collection/applogs123`).

> **Note**
>
> Las claves de condiciones de colección no se aplican a nivel de índice. Por ejemplo, en la política anterior, la condición no se aplicaría a una política de acceso o seguridad que contenga la cadena de recursos`index/logs-application/*`.

Para obtener una lista de las claves de condición de OpenSearch sin servidor, consulte Claves de condición de Amazon OpenSearch sin servidor en la Referencia de autorizaciones de servicio. Para obtener más información sobre las acciones y los recursos con los que puede utilizar una clave de condición, consulte Acciones definidas por Amazon OpenSearch sin servidor.

ABAC con OpenSearch sin servidor

Admite ABAC (etiquetas en las políticas): sí

El control de acceso basado en atributos (ABAC) es una estrategia de autorización que define permisos en función de atributos denominados etiquetas. Puede asociar etiquetas a entidades de IAM y recursos de AWS y, a continuación, diseñar políticas de ABAC para permitir operaciones cuando la etiqueta de la entidad principal coincida con la etiqueta del recurso.

Para controlar el acceso en función de etiquetas, debe proporcionar información de las etiquetas en el elemento de condición de una política utilizando las claves de condición `aws:ResourceTag/key-name`, `aws:RequestTag/key-name` o `aws:TagKeys`.

Si un servicio admite las tres claves de condición para cada tipo de recurso, el valor es Sí para el servicio. Si un servicio admite las tres claves de condición solo para algunos tipos de recursos, el valor es Parcial.

Para obtener más información sobre ABAC, consulte Definición de permisos con la autorización de ABAC en la Guía del usuario de IAM. Para ver un tutorial con los pasos para configurar ABAC, consulte Uso del control de acceso basado en atributos (ABAC) en la Guía del usuario de IAM.

Para obtener más información sobre el etiquetado de recursos OpenSearch sin servidor, consulte the section called “Etiquetado de colecciones”.

Uso de credenciales temporales con OpenSearch sin servidor

Compatibilidad con credenciales temporales: sí

Las credenciales temporales proporcionan acceso a corto plazo a los recursos de AWS y se crean automáticamente cuando se utiliza la federación o se cambia de rol. AWS recomienda que genere de forma dinámica credenciales temporales en lugar de usar claves de acceso a largo plazo. Para obtener más información, consulte Credenciales de seguridad temporales en IAM y Servicios de AWS que funcionan con IAM en la Guía del usuario de IAM.

Uso de roles vinculados a servicios para OpenSearch sin servidor

Admite roles vinculados a servicios: sí

Un rol vinculado a servicios es un tipo de rol de servicio que está vinculado a un Servicio de AWS. El servicio puedes asumir el rol para realizar una acción en su nombre. Los roles vinculados a servicios aparecen en la Cuenta de AWS y son propiedad del servicio. Un administrador de IAM puede ver, pero no editar, los permisos de los roles vinculados a servicios.

Para obtener más información sobre cómo crear o administrar roles vinculados a servicios de OpenSearch sin servidor, consulte the section called “Rol de creación de colecciones”.

Otros tipos de políticas

AWS admite otros tipos de políticas adicionales menos frecuentes. Estos tipos de políticas pueden establecer el máximo de permisos que los tipos de políticas más frecuentes le conceden.

- Políticas de control de servicio (SCP): las SCP son políticas de JSON que especifican los permisos máximos de una organización o una unidad organizativa (OU) en AWS Organizations. AWS Organizations es un servicio que le permite agrupar y administrar de manera centralizada varias cuentas de AWS que posea su empresa. Si habilita todas las funciones en una organización, entonces podrá aplicar políticas de control de servicio (SCP) a una o todas sus cuentas. Las SCP limitan los permisos de las entidades de las cuentas miembro, incluido cada usuario raíz de la cuenta de AWS. Para obtener más información acerca de SCP y Organizations, consulta Políticas de control de servicios en la Guía del usuario de AWS Organizations.
- Políticas de control de recursos (RCP): las RCP son políticas JSON que permiten establecer los permisos máximos disponibles para los recursos de las cuentas sin actualizar las políticas de IAM asociadas a cada recurso que posea. La RCP limita los permisos de los recursos en las cuentas de miembros y puede afectar a los permisos efectivos de las identidades, incluidos los usuarios raíz de la cuenta de AWS, independientemente de si pertenecen a su organización. Para obtener más información sobre Organizations y RCP, incluida una lista de los servicios de AWS que admiten RCP, consulte Políticas de control de recursos (RCP) en la Guía del usuario de AWS Organizations.

Ejemplos de políticas basadas en identidades para OpenSearch sin servidor

De forma predeterminada, los usuarios y roles no tienen permiso para crear, ver ni modificar recursos de OpenSearch sin servidor. Un administrador de IAM puede crear políticas de IAM para conceder permisos a los usuarios para realizar acciones en los recursos que necesitan.

Para obtener información acerca de cómo crear una política basada en identidades de IAM mediante el uso de estos documentos de políticas JSON de ejemplo, consulte Creación de políticas de IAM (consola) en la Guía del usuario de IAM.

Para obtener más información sobre las acciones y los tipos de recursos definidos por Amazon OpenSearch sin servidor, incluido el formato de los ARN para cada uno de los tipos de recursos, consulte Acciones, recursos y claves de condición para Amazon OpenSearch sin servidor en la Referencia de autorizaciones de servicio.

Temas

- Prácticas recomendadas sobre las políticas
- Uso de OpenSearch sin servidor en la consola
- Administración de colecciones de OpenSearch sin servidor
- Visualización de colecciones de OpenSearch sin servidor
- Uso de la operaciones de la API de OpenSearch
- ABAC para operaciones de la API de OpenSearch

Prácticas recomendadas sobre las políticas

Las políticas basadas en identidad son muy eficaces. Determinan si alguien puede crear, acceder o eliminar los recursos de OpenSearch sin servidor de su cuenta. Estas acciones pueden generar costos adicionales para su Cuenta de AWS. Siga estas directrices y recomendaciones al crear o editar políticas basadas en identidades:

Las políticas basadas en identidades determinan si alguien puede crear, acceder o eliminar recursos de OpenSearch sin servidor de la cuenta. Estas acciones pueden generar costos adicionales para su Cuenta de AWS. Siga estas directrices y recomendaciones al crear o editar políticas basadas en identidades:

- Comienza con las políticas administradas por AWSy continúa con los permisos de privilegio mínimo: a fin de comenzar a conceder permisos a los usuarios y las cargas de tarea, utiliza las políticas administradas por AWS, que conceden permisos para muchos casos de uso comunes.

Están disponibles en su Cuenta de AWS. Se recomienda definir políticas gestionadas por el cliente de AWS específicas para sus casos de uso a fin de reducir aún más los permisos. Con el fin de obtener más información, consulte las políticas administradas por AWS o las políticas administradas por AWS para funciones de tarea en la Guía de usuario de IAM.

- Aplique permisos de privilegio mínimo: cuando establezca permisos con políticas de IAM, conceda solo los permisos necesarios para realizar una tarea. Para ello, debe definir las acciones que se pueden llevar a cabo en determinados recursos en condiciones específicas, también conocidos como permisos de privilegios mínimos. Con el fin de obtener más información sobre el uso de IAM para aplicar permisos, consulte Políticas y permisos en IAM en la Guía del usuario de IAM.
- Utilice condiciones en las políticas de IAM para restringir aún más el acceso: puede agregar una condición a sus políticas para limitar el acceso a las acciones y los recursos. Por ejemplo, puedes escribir una condición de políticas para especificar que todas las solicitudes deben enviarse utilizando SSL. También puedes usar condiciones para conceder acceso a acciones de servicios si se emplean a través de un Servicio de AWS determinado como, por ejemplo, CloudFormation. Para obtener más información, consulte Elementos de la política de JSON de IAM: Condición en la Guía del usuario de IAM.
- Utiliza el analizador de acceso de IAM para validar las políticas de IAM con el fin de garantizar la seguridad y funcionalidad de los permisos: el analizador de acceso de IAM valida políticas nuevas y existentes para que respeten el lenguaje (JSON) de las políticas de IAM y las prácticas recomendadas de IAM. El analizador de acceso de IAM proporciona más de 100 verificaciones de políticas y recomendaciones procesables para ayudar a crear políticas seguras y funcionales. Para más información, consulte Validación de políticas con el Analizador de acceso de IAM en la Guía del usuario de IAM.
- Solicite la autenticación multifactor (MFA): si se encuentra en una situación en la que necesite usuarios raíz o de IAM en su Cuenta de AWS, active la MFA para obtener una mayor seguridad. Para exigir la MFA cuando se invoquen las operaciones de la API, añada condiciones de MFA a sus políticas. Para más información, consulte Acceso seguro a la API con MFA en la Guía del usuario de IAM.

Para obtener más información sobre las prácticas recomendadas de IAM, consulte Prácticas recomendadas de seguridad en IAM en la Guía del usuario de IAM.

Uso de OpenSearch sin servidor en la consola

Para acceder a OpenSearch sin servidor en la consola de OpenSearch Service, debe tener un conjunto mínimo de permisos. Estos permisos deben permitirle registrar y consultar los detalles

acerca de los recursos de OpenSearch sin servidor en su cuenta de AWS. Si se crea una política basada en identidades que es más restrictiva que los permisos necesarios mínimos, la consola no funcionará del modo esperado para las entidades (tales como roles de IAM) que tengan esa política.

No es necesario que conceda permisos mínimos para la consola a los usuarios que solo realizan llamadas a la AWS CLI o a la API de AWS. En su lugar, permite acceso únicamente a las acciones que coincidan con la operación de API que intenta realizar.

La siguiente política permite al usuario acceder a OpenSearch sin servidor desde la consola de OpenSearch Service:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Resource": "*",
            "Effect": "Allow",
            "Action": [
                "aoss:ListCollections",
                "aoss:BatchGetCollection",
                "aoss:ListAccessPolicies",
                "aoss:ListSecurityConfigs",
                "aoss:ListSecurityPolicies",
                "aoss:ListTagsForResource",
                "aoss:ListVpcEndpoints",
                "aoss:GetAccessPolicy",
                "aoss:GetAccountSettings",
                "aoss:GetSecurityConfig",
                "aoss:GetSecurityPolicy"
            ]
        }
    ]
}
```

Administración de colecciones de OpenSearch sin servidor

Esta política es un ejemplo de la política de “administración de colecciones” que permite al usuario gestionar y administrar las colecciones de Amazon OpenSearch sin servidor. El usuario puede crear, ver y eliminar colecciones.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Resource": "arn:aws:aoss:us-east-1:111122223333:collection/*",
            "Action": [
                "aoss:CreateCollection",
                "aoss:DeleteCollection",
                "aoss:UpdateCollection"
            ],
            "Effect": "Allow"
        },
        {
            "Resource": "*",
            "Action": [
                "aoss:BatchGetCollection",
                "aoss:ListCollections",
                "aoss:CreateAccessPolicy",
                "aoss:CreateSecurityPolicy"
            ],
            "Effect": "Allow"
        }
    ]
}
```

Visualización de colecciones de OpenSearch sin servidor

Este ejemplo de política permite a un usuario ver los detalles de todas las colecciones de Amazon OpenSearch sin servidor de su cuenta. El usuario no puede modificar las colecciones ni las políticas de seguridad asociadas.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Resource": "*",
```

```
            "Action": [
                "aoss:ListAccessPolicies",
                "aoss:ListCollections",
                "aoss:ListSecurityPolicies",
                "aoss:ListTagsForResource",
                "aoss:BatchGetCollection"
            ],
            "Effect": "Allow"
        }
    ]
}
```

Uso de la operaciones de la API de OpenSearch

Las operaciones de la API del plano de datos consisten en las funciones que se utilizan en OpenSearch sin servidor para derivar el valor del servicio en tiempo real. Las operaciones de la API del plano de control consisten en las funciones que se utilizan para configurar el entorno.

Para acceder a las API del plano de datos de Amazon OpenSearch sin servidor y a OpenSearch Dashboards desde el navegador, debe añadir dos permisos de IAM para los recursos de recopilación. Estos permisos son `aoss:APIAccessAll` y `aoss:DashboardsAccessAll`.

> **Note**
>
> A partir del 10 de mayo de 2023, OpenSearch sin servidor requerirá estos dos nuevos permisos de IAM para los recursos de recopilación. El permiso `aoss:APIAccessAll` concede el acceso al plano de datos y el permiso `aoss:DashboardsAccessAll` permite abrir OpenSearch Dashboards desde el navegador. Si no se agregan los dos nuevos permisos de IAM, se produce un error 403.

Este ejemplo de política permite a un usuario acceder a las API del plano de datos para una colección específica de su cuenta y acceder a OpenSearch Dashboards para todas las colecciones de su cuenta.

JSON

```
{
    "Version":"2012-10-17",
```

```
    "Statement": [
         {
            "Effect": "Allow",
            "Action": "aoss:APIAccessAll",
            "Resource": "arn:aws:aoss:us-
east-1:111122223333:collection/collection-id"
        },
        {
            "Effect": "Allow",
            "Action": "aoss:DashboardsAccessAll",
            "Resource": "arn:aws:aoss:us-east-1:111122223333:dashboards/default"
        }
    ]
}
```

Tanto `aoss:APIAccessAll` como `aoss:DashboardsAccessAll` otorgan el permiso completo de IAM a los recursos de la colección, mientras que el permiso de los paneles también permite el acceso a OpenSearch Dashboards. Cada permiso funciona de forma independiente, por lo que una denegación explícita de `aoss:APIAccessAll` no bloquea el acceso `aoss:DashboardsAccessAll` a los recursos, incluidas las herramientas para desarrolladores. Lo mismo sucede con una negación en `aoss:DashboardsAccessAll`. OpenSearch sin servidor admite las siguientes claves de condición globales:

- `aws:CalledVia`
- `aws:CalledViaAWSService`
- `aws:CalledViaFirst`
- `aws:CalledViaLast`
- `aws:CurrentTime`
- `aws:EpochTime`
- `aws:PrincipalAccount`
- `aws:PrincipalArn`
- `aws:PrincipallsAWSService`
- `aws:PrincipalOrgID`
- `aws:PrincipalOrgPaths`
- `aws:PrincipalType`
- `aws:PrincipalServiceName`

- aws:PrincipalServiceNamesList
- aws:ResourceAccount
- aws:ResourceOrgID
- aws:ResourceOrgPaths
- aws:RequestedRegion
- aws:ResourceTag
- aws:SourceIp
- aws:SourceVpce
- aws:SourceVpc
- aws:userid
- aws:username
- aws:VpcSourceIp

El siguiente es un ejemplo de cómo usar aws:SourceIp en el bloque de condición de la política de IAM de su principal para llamadas al plano de datos:

```
"Condition": {
    "IpAddress": {
         "aws:SourceIp": "203.0.113.0"
    }
}
```

El siguiente es un ejemplo de cómo usar aws:SourceVpc en el bloque de condición de la política de IAM de su principal para llamadas al plano de datos:

```
"Condition": {
    "StringEquals": {
        "aws:SourceVpc": "vpc-0fdd2445d8EXAMPLE"
    }
}
```

Además, se ofrece soporte para las siguientes claves específicas de OpenSearch sin servidor:

- aoss:CollectionId
- aoss:collection

El siguiente es un ejemplo de cómo usar `aoss:collection` en el bloque de condición de la política de IAM de su principal para llamadas al plano de datos:

```
"Condition": {
    "StringLike": {
         "aoss:collection": "log-*"
    }
}
```

ABAC para operaciones de la API de OpenSearch

Las políticas basadas en identidad le permiten usar etiquetas para controlar el acceso a las API del plano de datos de Amazon OpenSearch sin servidor. La siguiente política es un ejemplo para permitir que las entidades principales adjuntas accedan a las API del plano de datos si la colección tiene la etiqueta `team:devops`:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "aoss:APIAccessAll",
            "Resource": "arn:aws:aoss:us-
east-1:111122223333:collection/collection-id",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/team": "devops"
                }
            }
        }
    ]
}
```

La siguiente política es un ejemplo para denegar que las entidades principales adjuntas accedan a las API del plano de datos y al acceso a Dashboards si la colección tiene la etiqueta `environment:production`:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": [
                "aoss:APIAccessAll",
                "aoss:DashboardsAccessAll"
            ],
            "Resource": "arn:aws:aoss:us-
east-1:111122223333:collection/collection-id"
        }
    ]
}
```

Amazon OpenSearch sin servidor no admite las claves de condición global `RequestTag` y `TagKeys` para las API del plano de datos.

Compatibilidad de Centro de identidades de IAM con Amazon OpenSearch sin servidor

Compatibilidad de Centro de identidades de IAM con Amazon OpenSearch sin servidor

Puede usar entidades principales de Centro de identidades de IAM (usuarios y grupos) para acceder a los datos de Amazon OpenSearch sin servidor a través de Amazon OpenSearch Applications. Para habilitar la compatibilidad con Centro de identidades de IAM para Amazon OpenSearch sin servidor, deberá habilitar el uso de Centro de identidades de IAM. Para obtener más información sobre cómo hacerlo, consulte ¿Qué es Centro de identidades de IAM?

Después de crear la instancia de Centro de identidades de IAM, el administrador de la cuenta del cliente debe crear una aplicación de Centro de identidades de IAM para el servicio Amazon OpenSearch sin servidor. Esto se puede hacer llamando a CreateSecurityConfig:. El administrador de la cuenta del cliente puede especificar qué atributos se usarán para autorizar la solicitud. Los atributos predeterminados que se usan son `UserId` y `GroupId`..

La integración de Centro de identidades de IAM para Amazon OpenSearch sin servidor utiliza los siguientes permisos de Centro de identidades de IAM (IAM) de AWS:

- `aoss:CreateSecurityConfig`: crear un proveedor de Centro de identidades de IAM
- `aoss:ListSecurityConfig`: enumerar todos los proveedores de Centro de identidades de IAM en la cuenta actual.
- `aoss:GetSecurityConfig`: ver información del proveedor de Centro de identidades de IAM.
- `aoss:UpdateSecurityConfig`: modificar una configuración determinada de Centro de identidades de IAM
- `aoss:DeleteSecurityConfig`: eliminación de un proveedor de confianza de Centro de identidades de IAM.

La siguiente política de acceso basada en identidad se puede usar para administrar todas las configuraciones de Centro de identidades de IAM:

JSON

```
{
"Version": "2012-10-17",
    "Statement": [
        {
"Action": [
                "aoss:CreateSecurityConfig",
                "aoss:DeleteSecurityConfig",
                "aoss:GetSecurityConfig",
                "aoss:UpdateSecurityConfig",
                "aoss:ListSecurityConfigs"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

> **Note**
>
> El elemento `Resource` debe ser un comodín.

Crear un proveedor de Centro de identidades de IAM (consola)

Puede crear un proveedor de Centro de identidades de IAM para habilitar la autenticación con OpenSearch Application. Para habilitar la autenticación de Centro de identidades de IAM en OpenSearch Dashboards, realice los siguientes pasos:

1. Inicie sesión en la consola de Amazon OpenSearch Service.
2. En el panel de navegación izquierdo, expanda sin servidor y elija Autenticación.
3. Seleccione Autenticación de Centro de identidades de IAM.
4. Seleccione Editar.
5. Marque la casilla junto a Autenticarse con Centro de identidades de IAM.
6. Seleccione la clave de atributo de usuario y grupo del menú desplegable. Los atributos de usuario se usarán para autorizar a los usuarios según `UserName`, `UserId` y `Email`. Los atributos de grupo se usarán para autenticar a los usuarios según `GroupName` y `GroupId`.
7. Seleccione la instancia de Centro de identidades de IAM.
8. Seleccione Guardar.

Crear un proveedor de Centro de identidades de IAM (AWS CLI)

Para crear un proveedor de Centro de identidades de IAM usando el AWS Command Line Interface (AWS CLI), use el siguiente comando:

```
aws opensearchserverless create-security-config \
--region us-east-2 \
--name "iamidentitycenter-config" \
--description "description" \
--type "iamidentitycenter" \
--iam-identity-center-options '{
    "instanceArn": "arn:aws:sso:::instance/ssoins-99199c99e99ee999",
    "userAttribute": "UserName",
    "groupAttribute": "GroupId"
}'
```

Después de habilitar un Centro de identidades de IAM, los clientes solo pueden modificar los atributos de usuario y grupo.

```
aws opensearchserverless update-security-config \
--region us-east-1 \
```

```
--id <id_from_list_security_configs> \
--config-version <config_version_from_get_security_config> \
--iam-identity-center-options-updates '{
    "userAttribute": "UserId",
    "groupAttribute": "GroupId"
}'
```

Para ver el proveedor de Centro de identidades de IAM usando el AWS Command Line Interface, use el siguiente comando:

```
aws opensearchserverless list-security-configs --type iamidentitycenter
```

Eliminar un proveedor de Centro de identidades de IAM

El IAM Identity Center ofrece dos instancias de proveedores, una para su cuenta de organización y otra para su cuenta de miembro. Si necesita cambiar su instancia del Centro de identidades de IAM, debe eliminar su configuración de seguridad a través de la API de `DeleteSecurityConfig` y crear una nueva configuración de seguridad usando la nueva instancia del Centro de identidades de IAM. El siguiente comando se puede usar para eliminar un proveedor de Centro de identidades de IAM:

```
aws opensearchserverless delete-security-config \
--region us-east-1 \
--id <id_from_list_security_configs>
```

Conceder acceso de Centro de identidades de IAM a los datos de la colección

Después de habilitar su proveedor del Centro de identidades de IAM, puede actualizar la política de acceso a los datos de la colección para incluir entidades principales del Centro de identidades de IAM. Las entidades principales de Centro de identidades de IAM deben actualizarse en el siguiente formato:

```
[
   {
"Rules":[
       ...
      ],
      "Principal":[
         "iamidentitycenter/<iamidentitycenter-instance-id>/user/<UserName>",
         "iamidentitycenter/<iamidentitycenter-instance-id>/group/<GroupId>"
      ]
   }
```

```
]
```

> **Note**
>
> Amazon OpenSearch sin servidor admite solo una instancia de Centro de identidades de IAM para todas las colecciones de clientes y puede admitir hasta 100 grupos para un solo usuario. Si intenta usar más de la cantidad de instancias permitidas, experimentará inconsistencias con el procesamiento de autorización de la política de acceso a datos y recibirá un mensaje de error `403`.

Puede concederles acceso a colecciones, índices o ambos. Si desea que diferentes usuarios tengan distintos permisos, deberá crear múltiples reglas. Para obtener una lista de permisos disponibles, consulte Identidad y Administración de Acceso en Amazon OpenSearch Service. Para obtener información sobre cómo formatear una política de acceso, consulte Conceder acceso a identidades SAML a los datos de la colección.

IAM Identity Center ofrece dos instancias de proveedores, una para su cuenta de organización y otra para su cuenta miembro. Si necesita cambiar su instancia del Centro de identidades de IAM, debe eliminar su configuración de seguridad a través de la API de `DeleteSecurityConfig` y crear una nueva configuración de seguridad usando la nueva instancia del Centro de identidades de IAM. El siguiente comando se puede usar para eliminar un proveedor de Centro de identidades de IAM:

```
aws opensearchserverless delete-security-config \
--region us-east-1 \
--id <id_from_list_security_configs>
```

Cifrado en Amazon OpenSearch sin servidor

Cifrado en reposo

Cada colección de Amazon OpenSearch sin servidor que se crea está protegida con cifrado de datos en reposo, una característica de seguridad que ayuda a prevenir el acceso no autorizado a los datos. El cifrado en reposo utiliza AWS Key Management Service (AWS KMS) para almacenar y administrar las claves de cifrado. Utiliza el algoritmo estándar de cifrado avanzado con claves de 256 bits (AES-256) para realizar el cifrado.

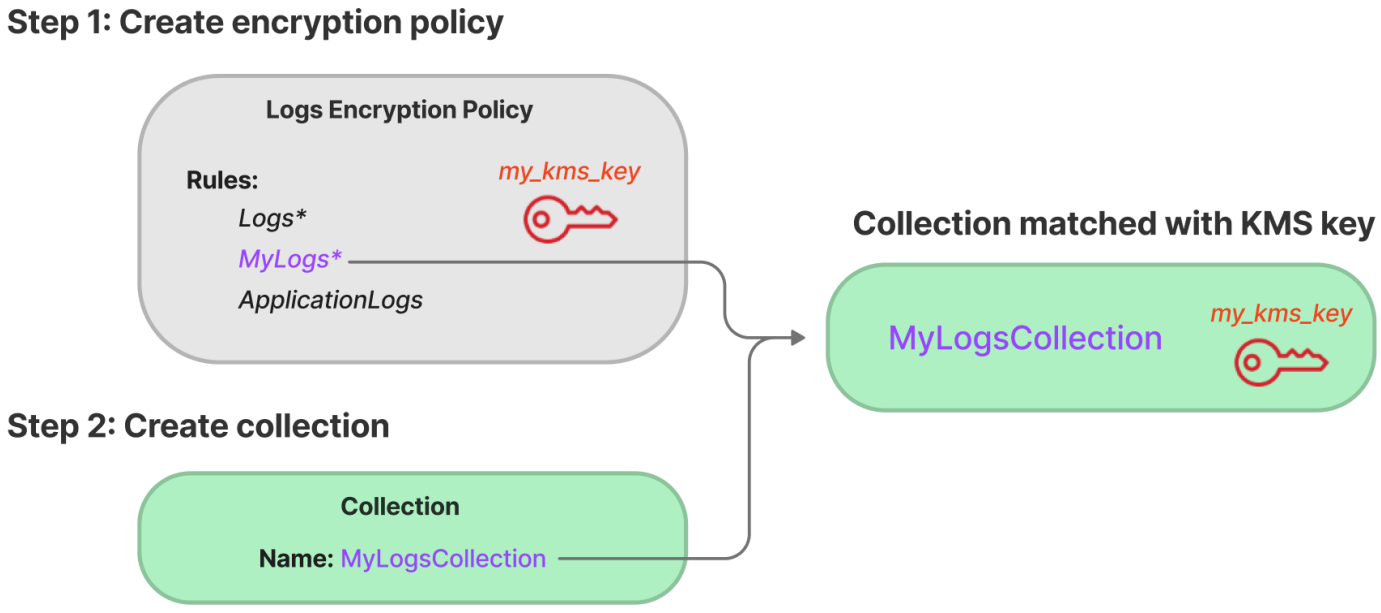
Temas

- Políticas de cifrado
- Consideraciones
- Permisos necesarios
- Política de claves para una clave administrada por el cliente
- Cómo OpenSearch sin servidor utiliza las concesiones en AWS KMS
- Creación de políticas de cifrado (consola)
- Creación de políticas de cifrado (AWS CLI)
- Visualización de políticas de cifrado
- Actualización de políticas de cifrado
- Eliminación de políticas de cifrado

Políticas de cifrado

Con las políticas de cifrado, puede administrar numerosas colecciones a escala mediante la asignación automática de una clave de cifrado a las colecciones más recientes que coincidan con un nombre o un patrón específicos.

Al crear una política de cifrado, puede especificar un prefijo, que es una regla de coincidencia basada en caracteres comodín, como `MyCollection*`, o escribir un nombre de colección único. Después, cuando se cree una colección que coincida con ese patrón de nombre o prefijo, se le asignarán de forma automática la política y la clave de KMS correspondientes.



Las políticas de cifrado contienen los siguientes elementos:

- `Rules`: una o más reglas de coincidencia de colecciones, cada una con los siguientes subelementos:
  - `ResourceType`: en la actualidad, la única opción es “colección”. Las políticas de cifrado solo se aplican a los recursos de colecciones.
  - `Resource`: uno o más nombres o patrones de colección a los que se aplicará la política, en el formato `collection/<collection name|pattern>`.
- `AWSOwnedKey`: si se utiliza una Clave propiedad de AWS o no.
- `KmsARN`: si configura el valor `AWSOwnedKey` en falso, especifique el nombre de recurso de Amazon (ARN) de la clave KMS con la que se va a cifrar las colecciones asociadas. Si incluye este parámetro, OpenSearch sin servidor ignora el parámetro `AWSOwnedKey`.

El siguiente ejemplo de política asignará una clave administrada por el cliente a cualquier colección futura denominada `autopartsinventory`, así como a las colecciones que comiencen por el término “ventas”:

```
{
   "Rules":[
      {
         "ResourceType":"collection",
         "Resource":[
            "collection/autopartsinventory",
            "collection/sales*"
         ]
      }
   ],
   "AWSOwnedKey":false,
   "KmsARN":"arn:aws:encryption:us-east-1:123456789012:key/93fd6da4-a317-4c17-bfe9-382b5d988b36"
}
```

Incluso si una política coincide con el nombre de una colección, puede optar por anular esta asignación automática durante la creación de la colección si el patrón de recursos contiene un comodín (*). Si decide anular la asignación automática de claves, OpenSearch sin servidor crea una política de cifrado denominada auto-**<collection-name>** y la adjunta a la colección. En un principio, la política solo se aplica a una única colección, pero se puede modificar para que incluya colecciones adicionales.

Si modifica las reglas de políticas para que dejen de coincidir con una colección, la clave KMS asociada continuará asignada a esa colección. La colección siempre permanece cifrada con su clave de cifrado inicial. Si desea cambiar la clave de cifrado de una colección, debe volver a crear la colección.

Si las reglas de varias políticas coinciden con una colección, se utiliza la regla más específica. Por ejemplo, si una política contiene una regla para `collection/log*` y otra para `collection/logSpecial`, se utiliza la clave de cifrado de la segunda política porque es más específica.

No puede usar un nombre o un prefijo en una política si ya existe en otra política. OpenSearch sin servidor muestra un error si intenta configurar patrones de recursos idénticos en diferentes políticas de cifrado.

Consideraciones

Cuando configure el cifrado de las colecciones, tenga en cuenta lo siguiente:

- El cifrado en reposo es obligatorio para todas las colecciones sin servidor.
- Puede elegir entre utilizar una clave administrada por el cliente o una Clave propiedad de AWS. Si elige una clave administrada por el cliente, le recomendamos habilitar la rotación automática de claves.
- No se puede cambiar la clave de cifrado de una colección después de su creación. Seleccione cuidadosamente qué AWS KMS usará la primera vez que configure una colección.
- Una colección solo puede coincidir con una única política de cifrado.
- Las colecciones con claves KMS únicas no pueden compartir las unidades de cómputo de OpenSearch (OCU) con otras colecciones. Cada colección con una clave única requiere 4 OCU propias.
- Si actualiza la clave KMS de una política de cifrado, el cambio no afectará a las colecciones coincidentes existentes con claves KMS ya asignadas.
- OpenSearch sin servidor no comprueba de forma explícita los permisos de usuario en las claves administradas por el cliente. Si un usuario tiene permisos para acceder a una colección mediante una política de acceso a los datos, podrá incorporar y consultar los datos cifrados con la clave asociada.

Permisos necesarios

El cifrado en reposo para OpenSearch sin servidor utiliza los siguientes permisos de AWS Identity and Access Management (IAM). Puede especificar las condiciones de IAM para restringir a los usuarios a colecciones específicas.

- `aoss:CreateSecurityPolicy`: cree una política de cifrado.
- `aoss:ListSecurityPolicies` enumere todas las políticas y colecciones de cifrado a las que están adjuntas.
- `aoss:GetSecurityPolicy`: consulte los detalles de una política de cifrado específica.
- `aoss:UpdateSecurityPolicy`: modifique una política de cifrado.
- `aoss:DeleteSecurityPolicy`: elimine una política de cifrado.

El siguiente ejemplo de política de acceso basada en la identidad proporciona los permisos mínimos necesarios para que un usuario administre las políticas de cifrado con el patrón de recursos `collection/application-logs`.

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "aoss:CreateSecurityPolicy",
            "aoss:UpdateSecurityPolicy",
            "aoss:DeleteSecurityPolicy",
            "aoss:GetSecurityPolicy"
         ],
         "Resource":"*",
         "Condition":{
            "StringEquals":{
               "aoss:collection":"application-logs"
            }
         }
      },
      {
         "Effect":"Allow",
```

```
         "Action":[
            "aoss:ListSecurityPolicies"
         ],
         "Resource":"*"
      }
   ]
}
```

Política de claves para una clave administrada por el cliente

Si selecciona una clave administrada por el cliente para proteger una colección, OpenSearch sin servidor obtiene permiso para utilizar la clave KMS en nombre de la entidad principal que realiza la selección. Esa entidad principal, un usuario o un rol, debe tener los permisos en la clave KMS que OpenSearch sin servidor necesita. Puede proporcionar estos permisos en una política de claves o en una política de IAM.

Como mínimo, OpenSearch sin servidor necesita los siguientes permisos en una clave administrada por el cliente:

- kms:DescribeKey
- kms:CreateGrant

Por ejemplo:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "kms:DescribeKey",
        "kms:CreateGrant"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "kms:ViaService": "aoss.us-east-1.amazonaws.com"
        },
```

```
        "Bool": {
          "kms:GrantIsForAWSResource": "true"
        }
      }
    }
  ]
}
```

OpenSearch sin servidor crea una concesión con los permisos kms:GenerateDataKey y kms:Decrypt.

Para más información, consulte Uso de políticas de claves en AWS KMS en la Guía del desarrollador de AWS Key Management Service.

Cómo OpenSearch sin servidor utiliza las concesiones en AWS KMS

OpenSearch sin servidor requiere una concesión para poder utilizar una clave administrada por el cliente.

Cuando crea una política de cifrado en su cuenta con una clave nueva, OpenSearch sin servidor crea una concesión en su nombre mediante el envío de la solicitud CreateGrant a AWS KMS. Las concesiones en AWS KMS se utilizan para otorgar a OpenSearch sin servidor acceso a una clave KMS en una cuenta de cliente.

OpenSearch sin servidor necesita la concesión para utilizar la clave administrada por el cliente para las siguientes operaciones internas:

- Enviar solicitudes de DescribeKey a AWS KMS para comprobar que el ID de la clave simétrica administrada por el cliente proporcionado es válido.
- Enviar solicitudes de GenerateDataKey a la clave KMS para crear claves de datos para cifrar objetos.
- Enviar solicitudes de Decrypt a AWS KMS para descifrar las claves de datos cifradas para que puedan usarse para cifrar sus datos.

Puede revocar el acceso a la concesión o eliminar el acceso del servicio a la clave administrada por el cliente en cualquier momento. Si lo hace, OpenSearch sin servidor no podrá acceder a ninguno de los datos cifrados por la clave administrada por el cliente, lo que afectará a todas las operaciones que dependen de esos datos y provocará errores y fallos de `AccessDeniedException` en los flujos de trabajo asíncronos.

OpenSearch sin servidor retira las concesiones en un flujo de trabajo asíncrono cuando una determinada clave administrada por el cliente no está asociada a ninguna política o colección de seguridad.

Creación de políticas de cifrado (consola)

En una política de cifrado, se especifica una clave KMS y una serie de patrones de colección a los que se aplicará la política. A cualquier colección nueva que coincida con uno de los patrones definidos en la política se le asignará la clave KMS correspondiente al crear la colección. Recomendamos que cree las políticas de cifrado antes de empezar a crear las colecciones.

Para crear una política de cifrado de OpenSearch sin servidor

1. Abra la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. En el panel de navegación izquierdo, amplíe Sin servidor y seleccione Políticas de cifrado.
3. Seleccione Crear política de cifrado.
4. Escriba un nombre y una descripción para la política.
5. En Recursos, ingrese uno o más patrones de recursos para esta política de cifrado. Todas las colecciones recién creadas en la región y Cuenta de AWS actuales que coincidan con uno de los patrones se asignan de forma automática a esta política. Por ejemplo, si introduce `ApplicationLogs` (sin comodín) y luego crea una colección con ese nombre, la política y la clave KMS correspondientes se asignarán a esa colección.

   También puede proporcionar un prefijo como `Logs*`, que asigna la política a cualquier colección nueva cuyo nombre comience por `Logs`. Mediante el uso de comodines, puede administrar la configuración de cifrado de varias colecciones a escala.
6. En Cifrado, seleccione la clave KMS que se debe utilizar.
7. Seleccione Crear.

Siguiente paso: crear colecciones

Después de configurar una o más políticas de cifrado, puede empezar a crear colecciones que coincidan con las reglas definidas en esas políticas. Para obtener instrucciones, consulte the section called “Creación de colecciones”.

En el paso de Cifrado de la creación de la colección, OpenSearch sin servidor le informa que el nombre que ingresó coincide con el patrón definido en una política de cifrado y asigna de forma

automática la clave KMS correspondiente a esa colección. Si el patrón de recursos contiene un comodín (*), puede anular la coincidencia y seleccionar su propia clave.

Creación de políticas de cifrado (AWS CLI)

Para crear una política de cifrado mediante las operaciones de la API de OpenSearch sin servidor, especifique los patrones de recursos y una clave de cifrado en formato JSON. La solicitud CreateSecurityPolicy acepta políticas insertadas y archivos .json.

Las políticas de cifrado tienen el siguiente formato. El archivo de muestra `my-policy.json` coincidirá con cualquier colección futura con el nombre `autopartsinventory`, así como con cualquier colección cuyo nombre comience con `sales`.

```
{
   "Rules":[
      {
         "ResourceType":"collection",
         "Resource":[
            "collection/autopartsinventory",
            "collection/sales*"
         ]
      }
   ],
   "AWSOwnedKey":false,
   "KmsARN":"arn:aws:encryption:us-east-1:123456789012:key/93fd6da4-a317-4c17-bfe9-382b5d988b36"
}
```

Para usar una clave propiedad del servicio, configure la `AWSOwnedKey` como `true`:

```
{
   "Rules":[
      {
         "ResourceType":"collection",
         "Resource":[
            "collection/autopartsinventory",
            "collection/sales*"
         ]
      }
   ],
   "AWSOwnedKey":true
}
```

La siguiente solicitud crea la política de cifrado:

```
aws opensearchserverless create-security-policy \
    --name sales-inventory \
    --type encryption \
    --policy file://my-policy.json
```

A continuación, utilice la operación de la API CreateCollection para crear una o más colecciones que coincidan con uno de los patrones de recursos.

Visualización de políticas de cifrado

Antes de crear una colección, puede que desee obtener una vista previa de las políticas de cifrado existentes en su cuenta para ver cuál tiene un patrón de recursos que coincide con el nombre de su colección. La siguiente solicitud ListSecurityPolicies enumera todas las políticas de cifrado de su cuenta:

```
aws opensearchserverless list-security-policies --type encryption
```

La solicitud devuelve información sobre todas las políticas de cifrado configuradas. Utilice el contenido del elemento `policy` para ver las reglas de patrón que se definen en la política:

```
{
   "securityPolicyDetails": [
      {
         "createdDate": 1663693217826,
         "description": "Sample encryption policy",
         "lastModifiedDate": 1663693217826,
         "name": "my-policy",
         "policy": "{\"Rules\":[{\"ResourceType\":\"collection\",\"Resource\":
[\"collection/autopartsinventory\",\"collection/sales*\"]}],\"AWSOwnedKey\":true}",
         "policyVersion": "MTY2MzY5MzIxNzgyNl8x",
         "type": "encryption"
      }
   ]
}
```

Para ver información detallada sobre una política específica, incluida la clave KMS, utilice el comando GetSecurityPolicy.

Actualización de políticas de cifrado

Si actualiza la clave KMS en una política de cifrado, el cambio solo se aplica a las colecciones recién creadas que coincidan con el nombre o patrón configurado. No afecta a las colecciones existentes que ya tienen claves KMS asignadas.

Lo mismo sucede con las reglas de coincidencia de las políticas. Si agrega, modifica o elimina una regla, el cambio solo se aplica a las colecciones recién creadas. Las colecciones existentes no pierden la clave KMS asignada si modifica las reglas de una política para que ya no coincida con el nombre de una colección.

Para actualizar una política de cifrado en la consola de OpenSearch sin servidor, seleccione Políticas de cifrado, seleccione la política que desee modificar y seleccione Editar. Realice los cambios y elija Guardar.

Para actualizar una política de cifrado mediante la API de OpenSearch sin servidor, utilice la operación UpdateSecurityPolicy. La siguiente solicitud actualiza una política de cifrado con un nuevo documento JSON de política:

```
aws opensearchserverless update-security-policy \
    --name sales-inventory \
    --type encryption \
    --policy-version 2 \
    --policy file://my-new-policy.json
```

Eliminación de políticas de cifrado

Cuando se elimina una política de cifrado, las colecciones que estén utilizando en ese momento la clave KMS definida en la política no se ven afectadas. Para eliminar una política en la consola de OpenSearch sin servidor, seleccione la política y elija Eliminar.

También puede utilizar la operación DeleteSecurityPolicy:

```
aws opensearchserverless delete-security-policy --name my-policy --type encryption
```

Cifrado en tránsito

En OpenSearch sin servidor, todas las rutas de una colección se cifran en tránsito mediante Transport Layer Security 1.2 (TLS) y un cifrado AES-256 estándar del sector. El acceso a todas

las API y paneles de Opensearch también se realiza a través de TLS 1.2. TLS es un conjunto de protocolos criptográficos estándares del sector que se utilizan para cifrar la información que se intercambia a través de la red.

Acceso a la red para Amazon OpenSearch sin servidor

La configuración de red de una colección de Amazon OpenSearch sin servidor determina si se puede acceder a la colección a través de Internet desde redes públicas o si se debe acceder a ella de forma privada.

El acceso privado puede aplicarse a uno de los siguientes lugares o ambos:

- Puntos de conexión de VPC administrados por OpenSearch sin servidor
- Servicios de AWS compatibles, como Amazon Bedrock

Puede configurar el acceso a la red por separado para el punto de conexión de OpenSearch de una colección y su punto de conexión correspondiente a OpenSearch Dashboards.

El acceso a la red es el mecanismo de aislamiento que permite el acceso desde diferentes redes de origen. Por ejemplo, si el punto de conexión de OpenSearch Dashboards de una colección es de acceso público, pero el punto de conexión de la API de OpenSearch no lo es, un usuario podrá acceder a los datos de la colección únicamente a través de Dashboards cuando se conecte desde una red pública. Si el usuario intenta llamar directamente a las API de OpenSearch desde una red pública, se le bloqueará. La configuración de red se puede utilizar para estas permutaciones de origen a tipo de recurso. Amazon OpenSearch sin servidor admite la conectividad IPv4 e IPv6.

Temas

- Políticas de red
- Consideraciones
- Permisos necesarios para configurar políticas de red
- Prioridad política
- Creación de políticas de (consola)
- Creación de políticas de (AWS CLI)
- Visualización de políticas de red
- Actualización de las políticas de red

- Eliminar las políticas de red

Políticas de red

Las políticas de red le permiten administrar muchas colecciones a gran escala al asignar de forma automática la configuración de acceso a la red a las colecciones que cumplen con las reglas definidas en la política.

En una política de red, se especifican una serie de reglas. Estas reglas definen los permisos de acceso a los puntos de conexión de la colección y a los puntos de conexión de OpenSearch Dashboards. Cada regla consta de un tipo de acceso (público o privado) y un tipo de recurso (colección o punto de conexión de OpenSearch Dashboards). Para cada tipo de recurso (`collection` y `dashboard`), especifique una serie de reglas que definen a qué colecciones se aplicará la política.

En este ejemplo de política, la primera regla especifica el acceso al punto de conexión de VPC tanto en el punto de conexión de la colección como al punto de conexión de Dashboards para todas las colecciones que comiencen por el término `marketing*`. También especifica el acceso a Amazon Bedrock.

> **Note**
>
> El acceso privado a los Servicios de AWS como Amazon Bedrock se aplica solo al punto de conexión de OpenSearch de la colección, no al punto de conexión de OpenSearch Dashboards. Incluso si `ResourceType` es `dashboard`, no se puede conceder acceso a Servicios de AWS a OpenSearch Dashboards.

La segunda regla especifica el acceso público a la colección `finance`, pero solo para el punto de conexión de la colección (sin acceso a los paneles).

```
[
   {
      "Description":"Marketing access",
      "Rules":[
         {
            "ResourceType":"collection",
            "Resource":[
               "collection/marketing*"
```

```
            ]
         },
         {
            "ResourceType":"dashboard",
            "Resource":[
               "collection/marketing*"
            ]
         }
      ],
      "AllowFromPublic":false,
      "SourceVPCEs":[
         "vpce-050f79086ee71ac05"
      ],
      "SourceServices":[
         "bedrock.amazonaws.com"
      ],
   },
   {
      "Description":"Sales access",
      "Rules":[
         {
            "ResourceType":"collection",
            "Resource":[
               "collection/finance"
            ]
         }
      ],
      "AllowFromPublic":true
   }
]
```

Esta política proporciona acceso público solo a OpenSearch Dashboards para las colecciones que comiencen por “finanzas”. Cualquier intento de acceder directamente a la API de OpenSearch será infructuoso.

```
[
  {
    "Description": "Dashboards access",
    "Rules": [
      {
        "ResourceType": "dashboard",
        "Resource": [
          "collection/finance*"
```

```
        ]
      }
    ],
    "AllowFromPublic": true
  }
]
```

Las políticas de red pueden aplicarse a las colecciones existentes, así como a las colecciones futuras. Por ejemplo, puede crear una colección y, a continuación, crear una política de red con una regla que coincida con el nombre de la colección. No es necesario crear políticas de red para crear colecciones.

Consideraciones

Al configurar el acceso a la red para sus colecciones, tenga en cuenta lo siguiente:

- Si planea configurar el acceso al punto de conexión de VPC para una colección, debe crear antes al menos un punto de conexión de VPC administrado por OpenSearch sin servidor.
- El acceso privado a los Servicios de AWS se aplica solo al punto de conexión de OpenSearch de la colección, no al punto de conexión de OpenSearch Dashboards. Incluso si `ResourceType` es `dashboard`, no se puede conceder acceso a Servicios de AWS a OpenSearch Dashboards.
- Si se puede acceder a una colección desde redes públicas, también se puede acceder a ella desde todos los puntos de conexión de VPC administrados por OpenSearch sin servidor y todos los Servicios de AWS.
- Se pueden aplicar varias políticas de red a una sola colección. Para obtener más información, consulte the section called “Prioridad política”.

Permisos necesarios para configurar políticas de red

El acceso a la red de OpenSearch sin servidor utiliza los siguientes permisos de AWS Identity and Access Management (IAM): Puede especificar condiciones de IAM para restringir a los usuarios a las políticas de red asociadas con colecciones específicas.

- `aoss:CreateSecurityPolicy`: cree una política de acceso a la red.
- `aoss:ListSecurityPolicies`: enumere todas las políticas de red de la cuenta actual.
- `aoss:GetSecurityPolicy`: vea una especificación de la política de acceso a la red.
- `aoss:UpdateSecurityPolicy`: modifique una política de acceso a la red determinada y cambiar el ID de VPC o la designación de acceso público.

- `aoss:DeleteSecurityPolicy`: elimine una política de acceso a la red (después de separarla de todas las colecciones).

La siguiente política de acceso basada en identidades permite al usuario ver todas las políticas de red y actualizarlas según el patrón de recursos`collection/application-logs`:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "aoss:UpdateSecurityPolicy"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aoss:collection": "application-logs"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "aoss:ListSecurityPolicies",
                "aoss:GetSecurityPolicy"
            ],
            "Resource": "*"
        }
    ]
}
```

> **Note**
>
> Además, OpenSearch sin servidor requiere los permisos `aoss:APIAccessAll` y `aoss:DashboardsAccessAll` para los recursos de la colección. Para obtener más información, consulte the section called “Uso de la operaciones de la API de OpenSearch”.

Prioridad política

Puede haber situaciones en las que las reglas de la política de red se superpongan, dentro de las políticas o entre ellas. Cuando esto ocurre, una regla que especifica el acceso público anula la regla que especifica el acceso privado para cualquier colección que sea común a ambas reglas.

Por ejemplo, en la siguiente política, ambas reglas asignan acceso a la red de la colección `finance`, pero una regla especifica el acceso a la VPC y la otra especifica el acceso público. En esta situación, el acceso público anula el acceso a la VPC únicamente para la recaudación de fondos (porque existe en ambas reglas), por lo que se podrá acceder a la colección de fondos desde las redes públicas. La colección de ventas tendrá acceso a VPC desde el punto de conexión especificado.

```
[
   {
      "Description":"Rule 1",
      "Rules":[
         {
            "ResourceType":"collection",
            "Resource":[
               "collection/sales",
               "collection/finance"
            ]
         }
      ],
      "AllowFromPublic":false,
      "SourceVPCEs":[
         "vpce-050f79086ee71ac05"
      ]
   },
   {
      "Description":"Rule 2",
      "Rules":[
         {
            "ResourceType":"collection",
            "Resource":[
               "collection/finance"
            ]
         }
      ],
      "AllowFromPublic":true
   }
]
```

Si se aplican varios puntos de conexión de VPC de diferentes reglas a una colección, las reglas son aditivas y se podrá acceder a la colección desde todos los puntos de conexión especificados. Si configura `AllowFromPublic` como `true`, pero también proporciona uno o más `SourceVPCEs` o `SourceServices`, OpenSearch sin servidor ignora los puntos de conexión de VPC y los identificadores de servicio, y las colecciones asociadas tendrán acceso público.

Creación de políticas de (consola)

Las políticas de red se pueden aplicar tanto a las políticas existentes como a las políticas future. Se recomienda crear políticas de red antes de empezar a crear colecciones.

Para crear una política de red de OpenSearch sin servidor

1. Abra la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. En el panel de navegación de la izquierda, expanda Sin servidor y seleccione Políticas de red.
3. Seleccione Crear políticas de red.
4. Escriba un nombre y una descripción para la política.
5. Proporcione una o más reglas. Estas reglas definen los permisos de acceso para sus colecciones de OpenSearch sin servidor y sus puntos de conexión de OpenSearch Dashboards.

   Cada regla contiene los siguientes elementos:

Elemento	Descripción
Nombre de la regla	Nombre que describa el contenido de la regla. Por ejemplo, “Acceso a VPC para el equipo de marketing”.
Tipo de acceso	Seleccione el acceso público o privado. A continuación, seleccione una de las siguiente s opciones o ambas: • Puntos de conexión de VPC para el acceso: especifique uno o más puntos de conexión de VPC administrados por OpenSearch sin servidor.

Elemento	Descripción
	• Acceso privado al Servicio de AWS: seleccione uno o más Servicios de AWS compatibles.
Tipo de recurso	Seleccione si desea proporcionar acceso a los puntos de conexión de OpenSearch (los cuales permiten realizar llamadas a la API de OpenSearch), a OpenSearch Dashboards (que permiten el acceso a las visualizaciones y a la interfaz de usuario de los complemen tos de OpenSearch) o ambos. **Note** El acceso privado a los Servicio de AWS se aplica solo al punto de conexión de OpenSearch de la colección, no al punto de conexión de OpenSearch Dashboards. Incluso si selecciona OpenSearch Dashboards, solo se puede conceder acceso al punto de conexión a Servicios de AWS.

Para cada tipo de recurso que seleccione, puede elegir las colecciones existentes a las que aplicar la configuración de la política o crear uno o más patrones de recursos. Los patrones de recursos constan de un prefijo y un comodín (*) y definen a qué colecciones se aplicará la configuración de la política.

Por ejemplo, si incluye un patrón denominado `Marketing*`, a cualquier colección nueva o existente cuyo nombre comience por “Marketing” se le aplicará automáticamente la configuración de red de esta política. Un único comodín (*) aplica la política a todas las colecciones actuales y future.

Además, puede especificar el nombre de una colección futura sin un comodín, por ejemplo `Finance`. OpenSearch sin servidor aplicará la configuración de la política a cualquier colección recién creada con ese nombre exacto.

6. Cuando esté satisfecho, seleccione Crear.

Creación de políticas de (AWS CLI)

Para crear una política de red mediante las operaciones de la API sin servidor de OpenSearch, especifique las reglas en formato JSON. La solicitud CreateSecurityPolicy acepta políticas insertadas y archivos .json. Todas las colecciones y patrones deben tomar la forma `collection/<collection name|pattern>`.

> **Note**
>
> El tipo de recurso `dashboards` solo permite el acceso a OpenSearch Dashboards, pero para que este funcione, también debe permitir el acceso a la colección desde las mismas fuentes. Consulte la segunda política a continuación para ver un ejemplo.

Para especificar el acceso privado, incluya uno o los siguientes elementos:

- `SourceVPCEs`: especifique uno o más puntos de conexión de VPC administrados por OpenSearch sin servidor.
- `SourceServices`: especifique el identificador de uno o más Servicios de AWS compatibles. Actualmente, se admiten los siguientes identificadores de servicio:
  - `bedrock.amazonaws.com`: Amazon Bedrock

El siguiente ejemplo de política de red proporciona acceso privado, a una punto de conexión de VPC y Amazon Bedrock, a los puntos de conexión de las colecciones solo para las colecciones que comienzan con el prefijo `log*`. Los usuarios autenticados no pueden iniciar sesión en OpenSearch Dashboards; solo pueden acceder al punto de conexión de la colección mediante programación.

```
[
   {
      "Description":"Private access for log collections",
      "Rules":[
```

```
         {
            "ResourceType":"collection",
            "Resource":[
               "collection/log*"
            ]
         }
      ],
      "AllowFromPublic":false,
      "SourceVPCEs":[
         "vpce-050f79086ee71ac05"
      ],
      "SourceServices":[
         "bedrock.amazonaws.com"
      ],
   }
]
```

La siguiente política proporciona acceso público a los puntos de conexión de OpenSearch y a OpenSearch Dashboards para una sola colección denominada `finance`. Si la colección no existe, la configuración de red se aplicará a la colección siempre y cuando se cree.

```
[
   {
      "Description":"Public access for finance collection",
      "Rules":[
         {
            "ResourceType":"dashboard",
            "Resource":[
               "collection/finance"
            ]
         },
         {
            "ResourceType":"collection",
            "Resource":[
               "collection/finance"
            ]
         }
      ],
      "AllowFromPublic":true
   }
]
```

La siguiente solicitud crea la política de red anterior:

```
aws opensearchserverless create-security-policy \
    --name sales-inventory \
    --type network \
    --policy "[{\"Description\":\"Public access for finance collection\",\"Rules
\":[{\"ResourceType\":\"dashboard\",\"Resource\":[\"collection\/finance\"]},
{\"ResourceType\":\"collection\",\"Resource\":[\"collection\/finance\"]}],
\"AllowFromPublic\":true}]"
```

Para proporcionar la política en un archivo JSON, utilice el formato`--policy file://my-policy.json`

Visualización de políticas de red

Antes de crear una colección, puede que desee obtener una vista previa de las políticas de red existentes en su cuenta para ver cuáles tienen un patrón de recursos que coincide con el nombre de su colección. La siguiente solicitud de ListSecurityPolicies enumera todas las políticas de red de su cuenta:

```
aws opensearchserverless list-security-policies --type network
```

La solicitud devuelve información sobre todas las políticas de red configuradas. Para ver las reglas de patrón definidas en una política específica, busque la información de la política en el contenido del elemento `securityPolicySummaries` en la respuesta. Anote el `name` y `type` de esta política y utilice estas propiedades en una solicitud de GetSecurityPolicy para recibir una respuesta con los siguientes detalles de la política:

```
{
    "securityPolicyDetail": [
        {
            "type": "network",
            "name": "my-policy",
            "policyVersion": "MTY2MzY5MTY1MDA3Ml8x",
            "policy": "[{\"Description\":\"My network policy rule\",\"Rules\":
[{\"ResourceType\":\"dashboard\",\"Resource\":[\"collection/*\"]}],\"AllowFromPublic
\":true}]",
            "createdDate": 1663691650072,
            "lastModifiedDate": 1663691650072
        }
    ]
}
```

Para ver información detallada sobre una política específica, utilice el comando GetSecurityPolicy.

Actualización de las políticas de red

Al modificar los puntos de conexión de VPC o la designación de acceso público de una red, todas las colecciones asociadas se ven afectadas. Para actualizar una política de red en la consola de OpenSearch sin servidor, amplíe las políticas de red, seleccione la política que desee modificar y seleccione Editar. Realice los cambios y elija Guardar.

Para actualizar una política de red mediante la API OpenSearch sin servidor, utilice el comando UpdateSecurityPolicy. Debe incluir un control de versiones de política en la solicitud. Puede recuperar el control de versiones de la política mediante los comandos `ListSecurityPolicies` o `GetSecurityPolicy`. Incluir la versión más reciente de la política garantiza que no se anule inadvertidamente un cambio realizado por otra persona.

La siguiente solicitud actualiza una política de red con un nuevo documento JSON de política:

```
aws opensearchserverless update-security-policy \
    --name sales-inventory \
    --type network \
    --policy-version MTY2MzY5MTY1MDA3Ml8x \
    --policy file://my-new-policy.json
```

Eliminar las políticas de red

Para poder eliminar una política de red, debe desconectarla de todas las colecciones. Para eliminar una política en la consola de OpenSearch sin servidor, seleccione la política y elija Eliminar.

También puede utilizar el comando DeleteSecurityPolicy:

```
aws opensearchserverless delete-security-policy --name my-policy --type network
```

Cumplimiento FIPS en Amazon OpenSearch sin servidor

Amazon OpenSearch sin servidor admite el Estándar Federal de Procesamiento de Información (FIPS) 140-2, un estándar del Gobierno de EE. UU. y Canadá que especifica los requisitos de seguridad para módulos criptográficos que protegen información sensible. Al conectarse a puntos de conexión habilitados para FIPS con OpenSearch sin servidor, las operaciones criptográficas se realizan utilizando bibliotecas criptográficas validadas por FIPS.

Los puntos de conexión FIPS de OpenSearch sin servidor están disponibles en Regiones de AWS donde se admite FIPS. Estos puntos de conexión utilizan TLS 1.2 o una versión posterior y algoritmos criptográficos validados FIPS para todas las comunicaciones. Para obtener más información, consulte Cumplimiento FIPS en la Guía del usuario de AWS Verified Access.

Temas

- Uso de puntos de conexión FIPS con OpenSearch sin servidor
- Uso de puntos de conexión FIPS con SDK de AWS
- Configurar grupos de seguridad para puntos de conexión de VPC
- Uso del punto de conexión de VPC compatible con FIPS
- Verificar el cumplimiento de FIPS
- Solución de problemas de conectividad de puntos de conexión de FIPS en zonas alojadas privadas

Uso de puntos de conexión FIPS con OpenSearch sin servidor

En las Regiones de AWS donde se admite FIPS, las colecciones de OpenSearch sin servidor son accesibles tanto a través de puntos de conexión estándar como compatibles con FIPS. Para obtener más información, consulte Cumplimiento FIPS en la Guía del usuario de AWS Verified Access.

En los siguientes ejemplos, reemplace *collection_id* y *Región de AWS* con su ID de colección y su Región de AWS.

- Puntos de conexión estándar: **https://*collection_id*.*Región de AWS*.aoss.amazonaws.com**.
- Punto de conexión compatible con FIPS: **https://*collection_id*.*Región de AWS*.aoss-fips.amazonaws.com**.

De manera similar, OpenSearch Dashboards son accesibles tanto a través de puntos de conexión estándar como compatibles con FIPS:

- Punto de conexión estándar de Dashboards: **https://*collection_id*.*Región de AWS*.aoss.amazonaws.com/_dashboards**.
- Punto de conexión de Dashboards compatible con FIPS: **https://*collection_id*.*Región de AWS*.aoss-fips.amazonaws.com/_dashboards**.

> **Note**
>
> En las regiones habilitadas para FIPS, tanto los puntos de conexión estándar como los compatibles con FIPS proporcionan criptografía compatible con FIPS. Los puntos de conexión específicos de FIPS lo ayudan a cumplir con los requisitos de cumplimiento que exigen específicamente el uso de puntos de conexión con FIPS en el nombre.

Uso de puntos de conexión FIPS con SDK de AWS

Al usar SDK de AWS, puede especificar el punto de conexión FIPS al crear el cliente. En el siguiente ejemplo, reemplace *collection_id* y *Región de AWS* con su ID de colección y su Región de AWS.

```
# Python SDK example
from opensearchpy import OpenSearch, RequestsHttpConnection, AWSV4SignerAuth
import boto3
host = '"https://collection_id.Región de AWS.aoss-fips.amazonaws.com"
region = 'us-west-2'
service = 'aoss'
credentials = boto3.Session().get_credentials()
auth = AWSV4SignerAuth(credentials, region, service)
client = OpenSearch(
    hosts = [{'host': host, 'port': 443}],
    http_auth = auth,
    use_ssl = True,
    verify_certs = True,
    connection_class = RequestsHttpConnection,
    pool_maxsize = 20
)
```

Configurar grupos de seguridad para puntos de conexión de VPC

Para garantizar una comunicación adecuada con su punto de conexión de Amazon VPC compatible con FIPS, cree o modifique un grupo de seguridad para permitir tráfico HTTPS entrante (puerto TCP 443) desde los recursos de su VPC que necesiten acceder a OpenSearch sin servidor. Luego, asocie este grupo de seguridad con su punto de conexión de VPC durante la creación o modificando el punto de conexión después de la creación. Para obtener más información, consulte Crear un grupo de seguridad en la Guía del usuario de Amazon VPC.

Uso del punto de conexión de VPC compatible con FIPS

Después de crear el punto de conexión de VPC compatible con FIPS, puede usarlo para acceder a OpenSearch Serverless desde los recursos dentro de su VPC. Para usar el punto de conexión en operaciones de API, configure su SDK para usar el punto de conexión regional FIPS según se describe en la sección Uso de puntos de conexión FIPS con OpenSearch sin servidor. Para acceder a OpenSearch Dashboards, use la URL de Dashboards específica de la colección, que se enrutará automáticamente a través del punto de conexión de VPC compatible con FIPS cuando se acceda desde dentro de su VPC. Para obtener más información, consulte Utilización de OpenSearch Dashboards con Amazon OpenSearch Service.

Verificar el cumplimiento de FIPS

Para verificar que sus conexiones a OpenSearch sin servidor estén utilizando criptografía compatible con FIPS, use AWS CloudTrail para supervisar las llamadas a la API realizadas a OpenSearch sin servidor. Compruebe que el campo `eventSource` en los registros de CloudTrail muestre `aoss-fips.amazonaws.com` para las llamadas a la API.

Para acceder a OpenSearch Dashboards, puede usar las herramientas de desarrollo del navegador para inspeccionar los detalles de la conexión TLS y verificar que se estén utilizando conjuntos de cifrado compatibles con FIPS.

Solución de problemas de conectividad de puntos de conexión de FIPS en zonas alojadas privadas

Los puntos de conexión de conexión de FIPS funcionan con las colecciones de Amazon OpenSearch sin servidor que tienen acceso público. Para las nuevas colecciones de VPC que utilizan puntos de conexión de VPC recién creados, los puntos de conexión de FIPS funcionan según lo esperado. Para otras colecciones de VPC, es posible que deba realizar una configuración manual para garantizar que los puntos de conexión de FIPS funcionen correctamente.

Para configurar zonas alojadas privadas de FIPS en Amazon Route 53

1. Abra la consola de Route 53 en https://console.aws.amazon.com/route53/.
2. Revise sus zonas alojadas:
    a. Localice las zonas alojadas en las que se encuentran las Regiones de AWS de sus colecciones.
    b. Compruebe los patrones de nomenclatura de las zonas alojadas:

- Formato que no es de FIPS: *region*.aoss.amazonaws.com.
- Formato que es de FIPS: *region*.aoss-fips.amazonaws.com.

c. Confirme que el Tipo de todas sus zonas alojadas esté configurado como Zona alojada privada.

3. Si falta la zona alojada privada de FIPS:

a. Seleccione la zona alojada privada no FIPS correspondiente.

b. Copie la información de las VPC asociadas. Por ejemplo: `vpc-1234567890abcdef0 | us-east-2`.

c. Busque el registro de dominio con comodín. Por ejemplo: `*.us-east-2.aoss.amazonaws.com`.

d. Copie la información de Valor/Dirigir el tráfico a. Por ejemplo: .:`uoc1c1qsw7poexampleewjeno1pte3rw.3ym756xh7yj.aoss.searchservices.a`

4. Cree la zona alojada privada de FIPS:

a. Cree una nueva zona alojada privada con el formato de FIPS. Por ejemplo: `us-east-2.aoss-fips.amazonaws.com`.

b. En el caso de las VPC asociadas, ingrese la información de la VPC que ha copiado de la zona alojada privada que no es de FIPS.

5. Agregue una nueva regla con las siguientes opciones de configuración:

a. Nombre del registro: *

b. Tipo de registro: CNAME

c. Valor: ingrese la información de Valor/Dirigir el tráfico a que copió anteriormente.

Problemas comunes

Si tiene problemas de conectividad con los puntos de conexión de VPC compatibles con FIPS, consulte la siguiente información para ayudar a resolverlos.

- Errores de resolución de DNS: no puede resolver el nombre de dominio del punto de conexión de FIPS en su VPC
- Tiempos de espera de conexión: se agota el tiempo de espera de sus solicitudes al punto de conexión de FIPS

- Errores de acceso denegado: la autenticación o la autorización fallan cuando se utilizan puntos de conexión de FIPS
- Faltan registros de zonas alojadas privadas para colecciones exclusivas de VPC

Para solucionar problemas de conectividad de puntos de conexión de FIPS

1. Compruebe la configuración de su zona alojada privada:

   a. Confirme que existe una zona alojada privada para el dominio del punto de conexión de FIPS (`*.region.aoss-fips.amazonaws.com`).

   b. Compruebe que la zona alojada privada esté asociada a la VPC correcta.

      Para obtener más información, consulte Zonas alojadas privadas en la Guía para desarrolladores de Amazon Route 53 y Administración de nombres de DNS en la Guía de AWS PrivateLink.

2. Pruebe de la resolución de DNS:

   a. Conéctese a una instancia de EC2 en su VPC.

   b. Ejecuta el siguiente comando:

      ```
      nslookup collection-id.region.aoss-fips.amazonaws.com
      ```

   c. Compruebe que la respuesta incluya la dirección IP privada del punto de conexión de VPC.

      Para obtener más información, consulte Políticas de puntos de conexión y Atributos de DNS en la Guía del usuario de Amazon VPC.

3. Compruebe la configuración del grupo de seguridad:

   a. Compruebe que el grupo de seguridad adjunto al punto de conexión de VPC permita el tráfico HTTPS (puerto 443) desde sus recursos.

   b. Confirme que los grupos de seguridad de sus recursos permitan el tráfico saliente al punto de conexión de VPC.

   Para obtener más información, consulte Políticas de puntos de conexión en la Guía de AWS PrivateLink y Grupos de seguridad en la Guía del usuario de Amazon VPC.

4. Revise la configuración de la ACL de red:

- Compruebe que las ACL de red permiten el tráfico entre sus recursos y el punto de conexión de VPC.

  Para obtener más información, consulte la sección relacionada con las ACL de red en la Guía del usuario de Amazon VPC.

5. Revise su política de puntos de conexión:

   - Compruebe que la política de puntos de conexión de VPC permita las acciones necesarias en sus recursos de OpenSearch sin servidor.

     Para obtener más información, consulte Permisos de puntos de conexión de VPC necesarios y Políticas de puntos de conexión en la Guía de AWS PrivateLink.

> **Tip**
>
> Si utiliza solucionadores de DNS personalizados en su VPC, configúrelos para que reenvíen las solicitudes de dominios `*.amazonaws.com` a los servidores de AWS.

Control de acceso a los datos para Amazon OpenSearch sin servidor

Con el control de acceso a los datos de Amazon OpenSearch sin servidor, puede permitir que los usuarios accedan a las colecciones e índices, sin importar su origen de red o mecanismo de acceso. Puede proporcionar acceso a roles de IAM e identidades SAML.

Los permisos de acceso se administran mediante las políticas de acceso a los datos, que se aplican a las colecciones y a los recursos de índice. Las políticas de acceso a datos le ayudan a administrar las colecciones a escala mediante la asignación automática de permisos de acceso a las colecciones e índices que coinciden con un patrón específico. Se pueden aplicar varias políticas de acceso a los datos a un solo recurso. Tenga en cuenta que debe tener una política de acceso a los datos para su colección a fin de acceder a la URL de OpenSearch Dashboards.

Temas

- Políticas de acceso a datos frente a políticas de IAM
- Permisos de IAM necesarios para configurar las políticas de acceso a los datos
- Sintaxis de la política

- Permisos de política compatibles
- Ejemplos de conjuntos de datos en OpenSearch Dashboards
- Creación de políticas de acceso a los datos (consola)
- Creación de políticas de acceso a los datos (AWS CLI)
- Visualización de políticas de acceso a los datos
- Actualización de las políticas de acceso a los datos
- Eliminación de políticas de acceso a los datos
- Acceso a los datos entre cuentas

Políticas de acceso a datos frente a políticas de IAM

Las políticas de acceso a los datos están separadas de forma lógica de las políticas AWS Identity and Access Management (IAM). Los permisos de IAM controlan el acceso a las operaciones de la API sin servidor, como `CreateCollection` y `ListAccessPolicies`. Las políticas de acceso a los datos controlan el acceso a las operaciones de OpenSearch que admite OpenSearch sin servidor, como `PUT <index>` o `GET _cat/indices`.

Los permisos de IAM que controlan el acceso a las operaciones de la API de la política de acceso a los datos, como `aoss:CreateAccessPolicy` y `aoss:GetAccessPolicy` (que se describen en la siguiente sección), no afectan al permiso especificado en una política de acceso a los datos.

Por ejemplo, supongamos que una política de IAM niega a un usuario crear políticas de acceso a los datos para `collection-a`, pero le permite crear políticas de acceso a datos para todas las colecciones (`*`):

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": [
                "aoss:CreateAccessPolicy"
            ],
            "Resource": "*",
            "Condition": {
```

```
                "StringLike": {
                    "aoss:collection": "collection-a"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "aoss:CreateAccessPolicy"
            ],
            "Resource": "*"
        }
    ]
}
```

Si el usuario crea una política de acceso a los datos que permite ciertos permisos a todas las colecciones (`collection/*` o `index/*/*`), la política se aplicará a todas las colecciones, incluida la colección A.

> **Important**
>
> Contar con permisos dentro de una política de acceso a datos no es suficiente para acceder a los datos de la colección de OpenSearch sin servidor. A una entidad principal también se le debe conceder acceso a los permisos de IAM `aoss:APIAccessAll` y `aoss:DashboardsAccessAll`. Ambos permisos conceden acceso completo a los recursos de la colección, mientras que el permiso de los paneles también permite el acceso a OpenSearch Dashboards. Si una entidad principal no tiene estos dos permisos de IAM, recibirá errores 403 cuando intente enviar solicitudes a la colección. Para obtener más información, consulte the section called “Uso de la operaciones de la API de OpenSearch”.

Permisos de IAM necesarios para configurar las políticas de acceso a los datos

El control de acceso a los datos para OpenSearch sin servidor utiliza los siguientes permisos de IAM. Puede especificar las condiciones de IAM para restringir a los usuarios a nombres de políticas de acceso específicos.

- `aoss:CreateAccessPolicy`: cree una política de acceso.
- `aoss:ListAccessPolicies`: enumere todas las políticas de acceso.

- `aoss:GetAccessPolicy`: consulte los detalles sobre una política de acceso específica.
- `aoss:UpdateAccessPolicy`: modifique una política de acceso.
- `aoss:DeleteAccessPolicy`: elimine una política de acceso.

La siguiente política de acceso basada en la identidad permite al usuario ver todas las políticas de acceso y actualizar las políticas que contienen el patrón de recursos `collection/logs`.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Action": [
                "aoss:ListAccessPolicies",
                "aoss:GetAccessPolicy"
            ],
            "Effect": "Allow",
            "Resource": "*"
        },
        {
            "Action": [
                "aoss:UpdateAccessPolicy"
            ],
            "Effect": "Allow",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aoss:collection": [
                        "logs"
                    ]
                }
            }
        }
    ]
}
```

> **Note**
>
> Además, OpenSearch sin servidor requiere los permisos `aoss:APIAccessAll` y `aoss:DashboardsAccessAll` para los recursos de la colección. Para obtener más información, consulte the section called “Uso de la operaciones de la API de OpenSearch”.

Sintaxis de la política

Una política de acceso a datos incluye un conjunto de reglas, cada una con los siguientes elementos:

Elemento	Descripción
`ResourceType`	El tipo de recurso (colección o índice) al que se le aplican los permisos. Los permisos de alias y plantillas se encuentran en el nivel de la colección , mientras que los permisos para crear, modificar y buscar datos están en el nivel del índice. Para más información, consulte Permisos de políticas compatibles.
`Resource`	Una lista de nombres o patrones de recursos. Los patrones son prefijos seguidos de un comodín (*), que permiten que los permisos asociados se apliquen a varios recursos. • Las colecciones adoptan el formato `collection/` *`<name|pattern>`* . • Los índices toman el formato `index/`*`<collection-name|pattern>`* */`*`<index-name|pattern/>`* .
`Permission`	Una lista de permisos a otorgar para los recursos especificados. Para obtener una lista completa de las operaciones de la API y de los permisos que permiten, consulte the section called “Permisos y operaciones de la API de OpenSearch”.
`Principal`	Una lista de una o más entidades principales a las que conceder acceso. Las entidades principales pueden ser ARN de roles de IAM o identidades SAML. Estas entidades principales deben estar dentro de la Cuenta de AWS actual. Las políticas de acceso a los datos no admiten directamente el acceso entre cuentas, pero puede incluir en la política un rol que pueda asumir un usuario de otra Cuenta de AWS en la cuenta propietaria de la colección. Para

Elemento	Descripción
	obtener más información, consulte the section called “Acceso a los datos entre cuentas”.

La siguiente política de ejemplo otorga permisos de alias y plantillas a la colección llamada `autopartsinventory`, así como a cualquier colección que comience por el prefijo `sales*`. También otorga permisos de lectura y escritura a todos los índices dentro de la colección `autopartsinventory` y a todos los índices de la colección `salesorders` que comiencen por el prefijo `orders*`.

```
[
   {
      "Description": "Rule 1",
      "Rules":[
         {
            "ResourceType":"collection",
            "Resource":[
               "collection/autopartsinventory",
               "collection/sales*"
            ],
            "Permission":[
               "aoss:CreateCollectionItems",
               "aoss:UpdateCollectionItems",
               "aoss:DescribeCollectionItems"
            ]
         },
         {
            "ResourceType":"index",
            "Resource":[
               "index/autopartsinventory/*",
               "index/salesorders/orders*"
            ],
            "Permission":[
               "aoss:*"
            ]
         }
      ],
      "Principal":[
         "arn:aws:iam::123456789012:user/Dale",
         "arn:aws:iam::123456789012:role/RegulatoryCompliance",
```

```
        "saml/123456789012/myprovider/user/Annie",
        "saml/123456789012/anotherprovider/group/Accounting"
      ]
   }
]
```

No se puede denegar el acceso de forma explícita dentro de una política. Por lo tanto, todos los permisos de la política son aditivos. Por ejemplo, si una política concede el permiso `aoss:ReadDocument` a un usuario y otra `aoss:WriteDocument`, el usuario tendrá ambos permisos. Si una tercera política concede al mismo usuario el permiso `aoss:*`, el usuario puede realizar todas las acciones del índice asociado; los permisos más restrictivos no anulan los menos restrictivos.

Permisos de política compatibles

Las políticas de acceso a los datos admiten los siguientes permisos. Para conocer las operaciones de la API de OpenSearch que permite cada permiso, consulte the section called “Permisos y operaciones de la API de OpenSearch”.

Permisos de colección

- `aoss:CreateCollectionItems`
- `aoss:DeleteCollectionItems`
- `aoss:UpdateCollectionItems`
- `aoss:DescribeCollectionItems`
- `aoss:*`

Permisos de índice

- `aoss:ReadDocument`
- `aoss:WriteDocument`
- `aoss:CreateIndex`
- `aoss:DeleteIndex`
- `aoss:UpdateIndex`
- `aoss:DescribeIndex`
- `aoss:*`

Ejemplos de conjuntos de datos en OpenSearch Dashboards

OpenSearch Dashboards proporcionan conjuntos de datos de ejemplo que incluyen visualizaciones, paneles y otras herramientas para ayudarlo a explorar Dashboards antes de agregar sus propios datos. Para crear índices a partir de estos datos de ejemplo, necesita una política de acceso a los datos que otorgue permisos al conjunto de datos con el que desee trabajar. La siguiente política usa un comodín (*) para proporcionar permisos a los tres conjuntos de datos de ejemplo.

```
[
  {
    "Rules": [
      {
        "Resource": [
          "index/<collection-name>/opensearch_dashboards_sample_data_*"
        ],
        "Permission": [
          "aoss:CreateIndex",
          "aoss:DescribeIndex",
          "aoss:ReadDocument"
        ],
        "ResourceType": "index"
      }
    ],
    "Principal": [
      "arn:aws:iam::<account-id>:user/<user>"
    ]
  }
]
```

Creación de políticas de acceso a los datos (consola)

Puede crear una política de acceso a los datos con el editor visual o en el formato JSON. A cualquier colección nueva que coincida con uno de los patrones definidos en la política se le asignarán los permisos correspondientes al crear la colección.

Para crear una política de acceso a los datos de OpenSearch sin servidor

1. Abra la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. En el panel de navegación izquierdo, expanda sin servidor y, en Seguridad, elija Políticas de acceso a datos.
3. Seleccione Crear política de acceso.

4. Escriba un nombre y una descripción para la política.
5. Proporcione un nombre para la primera regla de la política. Por ejemplo, “Acceso a la colección de registros”.
6. Seleccione Agregar entidades principales, y luego uno o varios roles de IAM o usuarios y grupos de SAML para proporcionarles acceso a los datos.

   > **Note**
   >
   > Para seleccionar las entidades principales desde los menús desplegables, debe tener los permisos `iam:ListUsers` y `iam:ListRoles` (para las entidades principales de IAM) y el permiso `aoss:ListSecurityConfigs` (para las identidades de SAML).

7. Seleccione Otorgar y seleccione los permisos de alias, plantillas e índices que desea conceder a las entidades principales asociadas. Para obtener una lista completa de los permisos y el acceso que permiten, consulte the section called “Permisos y operaciones de la API de OpenSearch”.
8. (Opcional) Configure reglas adicionales para la política.
9. Seleccione Crear. Puede transcurrir alrededor de un minuto de retardo entre el momento en que se crea la política y el momento en el que se aplican los permisos. Si tarda más de 5 minutos, póngase en contacto con Soporte.

> **Important**
>
> Si la política solo incluye permisos de índices (y no permisos de colecciones), es posible que siga apareciendo un mensaje sobre las colecciones coincidentes que indique lo siguiente: `Collection cannot be accessed yet. Configure data access policies so that users can access the data within this collection`. Puede omitir esta advertencia. Las entidades principales autorizadas pueden seguir realizando las operaciones relacionadas con índices que tengan asignadas en la colección.

Creación de políticas de acceso a los datos (AWS CLI)

Para crear una política de acceso a los datos mediante la API de OpenSearch sin servidor, utilice el comando `CreateAccessPolicy`. El comando acepta políticas insertadas y archivos .json. Las políticas insertadas deben codificarse como una cadena de escape de JSON.

La siguiente solicitud crea una política de acceso a los datos:

```
aws opensearchserverless create-access-policy \
    --name marketing \
    --type data \
    --policy "[{\"Rules\":[{\"ResourceType\":\"collection\",\"Resource\":
[\"collection/autopartsinventory\",\"collection/sales*\"],\"Permission\":
[\"aoss:UpdateCollectionItems\"]},{\"ResourceType\":\"index\",\"Resource\":
[\"index/autopartsinventory/*\",\"index/salesorders/orders*\"],\"Permission
\":[\"aoss:ReadDocument\",\"aoss:DescribeIndex\"]}],\"Principal\":
[\"arn:aws:iam::123456789012:user/Shaheen\"]}]"
```

Para proporcionar la política dentro de un archivo .json, utilice el formato `--policy file://my-policy.json`.

Las entidades principales incluidas en la política ahora pueden usar las operaciones de OpenSearch a las que se les concedió acceso.

Visualización de políticas de acceso a los datos

Antes de crear una colección, puede que desee obtener una vista previa de las políticas de acceso a los datos existentes en la cuenta para ver cuáles tienen un patrón de recursos que coincide con el nombre de su colección. La siguiente solicitud de ListAccessPolicies enumera todas las políticas de acceso a los datos en su cuenta:

```
aws opensearchserverless list-access-policies --type data
```

La solicitud devuelve información sobre todas las políticas de acceso a los datos configuradas. Para ver las reglas de patrón definidas en una política específica, busque la información de la política en el contenido del elemento `accessPolicySummaries` en la respuesta. Anote el `name` y `type` de esta política y utilice estas propiedades en una solicitud de GetAccessPolicy para recibir una respuesta con los siguientes detalles de la política:

```
{
    "accessPolicyDetails": [
        {
            "type": "data",
            "name": "my-policy",
            "policyVersion": "MTY2NDA1NDE4MDg1OF8x",
            "description": "My policy",
            "policy": "[{\"Rules\":[{\"ResourceType\":\"collection\",
\"Resource\":[\"collection/autopartsinventory\",\"collection/sales*\"],
\"Permission\":[\"aoss:UpdateCollectionItems\"]},{\"ResourceType\":\"index\",
```

```
\"Resource\":[\"index/autopartsinventory/*\",\"index/salesorders/orders*\"],
\"Permission\":[\"aoss:ReadDocument\",\"aoss:DescribeIndex\"]}],\"Principal\":
[\"arn:aws:iam::123456789012:user/Shaheen\"]}]",
            "createdDate": 1664054180858,
            "lastModifiedDate": 1664054180858
        }
    ]
}
```

Puede incluir filtros de recursos para limitar los resultados a políticas que contengan colecciones o índices específicos:

```
aws opensearchserverless list-access-policies --type data --resource
 "index/autopartsinventory/*"
```

Para ver los detalles de una política específica, utilice el comando GetAccessPolicy.

Actualización de las políticas de acceso a los datos

Al actualizar una política de acceso a los datos, todas las colecciones asociadas se ven afectadas. Para actualizar una política de acceso a datos en la consola de OpenSearch sin servidor, seleccione Control de acceso a los datos, seleccione la política que desee modificar y seleccione Editar. Realice los cambios y elija Guardar.

Para actualizar una política de acceso a los datos mediante la API de OpenSearch sin servidor, envíe la solicitud `UpdateAccessPolicy`. Debe incluir una versión de la política, que puede recuperar mediante los comandos `ListAccessPolicies` or `GetAccessPolicy`. Incluir la versión más reciente de la política garantiza que no se anule inadvertidamente un cambio realizado por otra persona.

La siguiente solicitud de UpdateAccessPolicy actualiza una política de acceso a los datos con un nuevo documento de política JSON:

```
aws opensearchserverless update-access-policy \
    --name sales-inventory \
    --type data \
    --policy-version MTY2NDA1NDE4MDg1OF8x \
    --policy file://my-new-policy.json
```

Pueden transcurrir unos minutos de retardo entre el momento en que se actualiza la política y el momento en que se aplican los nuevos permisos.

Eliminación de políticas de acceso a los datos

Al eliminar una política de acceso a los datos, todas las colecciones asociadas pierden el acceso definido en la política. Asegúrese de que sus usuarios de IAM y SAML tengan el acceso adecuado a la colección antes de eliminar una política. Para eliminar una política en la consola de OpenSearch sin servidor, seleccione la política y elija Eliminar.

También puede utilizar el comando DeleteAccessPolicy:

```
aws opensearchserverless delete-access-policy --name my-policy --type data
```

Acceso a los datos entre cuentas

Si bien no puede crear una política de acceso a los datos con identidades o colecciones entre cuentas, sí puede configurar el acceso entre cuentas con la opción de asumir un rol. Por ejemplo, si la *account-a* es propietaria de una colección a la que la *account-b* necesita acceso, el usuario de la *account-b* puede asumir un rol en la *account-a*. El rol debe tener los permisos de IAM `aoss:APIAccessAll` y `aoss:DashboardsAccessAll` y estar incluido en la política de acceso a los datos de la *account-a*.

Acceda a Amazon OpenSearch sin servidor mediante un punto de conexión de la interfaz (AWS PrivateLink)

Puede utilizar AWS PrivateLink para crear una conexión privada entre la VPC y Amazon OpenSearch sin servidor. Puede acceder a OpenSearch sin servidor como si estuviera en su VPC, sin el uso de una puerta de enlace de Internet, un dispositivo NAT, una conexión VPN o una conexión Direct Connect. Las instancias de la VPC no necesitan direcciones IP públicas para acceder a OpenSearch sin servidor. Para obtener más información sobre el acceso a la red mediante VPC, consulte Network connectivity patterns for Amazon OpenSearch sin servidor.

Esta conexión privada se establece mediante la creación de un punto de conexión de interfaz alimentado por AWS PrivateLink . Creamos una interfaz de red de punto de conexión en cada subred especificada para el punto de conexión de interfaz. Se trata de interfaces de red administradas por el solicitante que sirven como punto de entrada para el tráfico destinado a OpenSearch sin servidor.

Para obtener más información, consulte Acceso a Servicios de AWS a través de AWS PrivateLink en la Guía de AWS PrivateLink.

Temas

- Resolución de DNS de los puntos de conexión de colección
- VPC y políticas de acceso a la red
- Políticas de punto de conexión y VPC
- Consideraciones
- Permisos necesarios
- Creación de un punto de conexión de interfaz para OpenSearch sin servidor
- Configuración de VPC compartida para Amazon OpenSearch sin servidor

Resolución de DNS de los puntos de conexión de colección

Al crear un punto de conexión de VPC, el servicio crea una nueva zona alojada privada Amazon Route 53 y la adjunta a la VPC. Esta zona alojada privada consta de un registro para resolver el registro DNS comodín de las colecciones de OpenSearch sin servidor (`*.aoss.us-east-1.amazonaws.com`) en las direcciones de interfaz utilizadas para el punto de conexión. Solo necesita un punto de conexión de VPC de OpenSearch sin servidor en una VPC para acceder a todas y cada una de las colecciones y paneles de control de cada Región de AWS. Cada VPC con un punto de conexión para OpenSearch sin servidor tiene su propia zona alojada privada adjunta.

OpenSearch sin servidor también crea un registro DNS comodín público de Route 53 para todas las colecciones de la región. El nombre DNS se resuelve en las direcciones IP públicas de OpenSearch sin servidor. Los clientes de las VPC que no tienen un punto de conexión de VPC sin servidor de OpenSearch o los clientes de redes públicas pueden usar el resolver público de Route 53 y acceder a las colecciones y los paneles de control con esas direcciones IP. El tipo de dirección IP (IPv4, IPv6 o Dualstack) del punto de conexión de VPC se determina en función de las subredes proporcionadas al crear un punto de conexión de interfaz para OpenSearch sin servidor.

> **Note**
>
> OpenSearch sin servidor crea una zona alojada privada adicional de Amazon Route 53 (`<region>.opensearch.amazonaws.com`) para la resolución de un dominio de OpenSearch Service. Puede actualizar el punto de conexión de VPC IPv4 existente a Dualstack mediante el comando update-vpc-endpoint de la AWS CLI.

La dirección de resolver de DNS de una VPC determinada es la segunda dirección IP del CIDR de la VPC. Todos los clientes de la VPC deben usar ese resolver para obtener la dirección de punto

de conexión de VPC para cualquier colección. El resolver utiliza una zona alojada privada creada por OpenSearch sin servidor. Basta con usar ese resolverr para todas las colecciones de cualquier cuenta. También es posible utilizar el resolver de VPC para algunos puntos de conexión de colección y el resolver público para otros, aunque normalmente no es necesario.

VPC y políticas de acceso a la red

Para conceder permisos de red a las API y los paneles de control de OpenSearch para sus colecciones, puede utilizar las políticas de acceso a la red de OpenSearch sin servidor. Puede controlar este acceso a la red desde sus puntos de conexión de VPC o desde la Internet pública. Como su política de red solo controla los permisos de tráfico, también debe configurar una política de acceso a los datos que especifique los permisos para operar con los datos de una colección y sus índices. Piense en un punto de conexión de VPC de OpenSearch sin servidor como un punto de acceso al servicio, una política de acceso a la red como el punto de acceso a nivel de red a las colecciones y los paneles de control, y una política de acceso a los datos como el punto de acceso para un control de acceso detallado para cualquier operación con los datos de la colección.

Como puede especificar varios ID de punto de conexión de VPC en una política de red, le recomendamos que cree un punto de conexión de VPC para cada VPC que necesite acceder a una colección. Estas VPC pueden pertenecer a cuentas AWS distintas de la cuenta propietaria de la política de red y colección de OpenSearch sin servidor. No recomendamos crear una solución de emparejamiento de VPC a VPC ni ninguna otra solución de proxy entre dos cuentas para que la VPC de una cuenta pueda utilizar el punto de conexión de VPC de otra cuenta. Esto es menos seguro y rentable que cada VPC que tenga su propio punto de conexión. El administrador de la otra VPC, que ha configurado el acceso al punto de conexión de esa VPC en la política de red, no podrá ver fácilmente la primera VPC.

Políticas de punto de conexión y VPC

Amazon OpenSearch sin servidor es compatible con las políticas del punto de conexión para las VPC. Una política de punto de conexión es una política basada en recursos de IAM que se puede asociar a un punto de conexión de VPC para controlar qué entidades principales de AWS pueden utilizar el punto de conexión para acceder a su servicio de AWS. Para obtener más información, consulte Controlar el acceso a puntos de conexión de VPC con políticas de punto de conexión.

Para usar una política de punto de conexión, primero debe crear un punto de conexión de interfaz. Puede crear un punto de conexión de interfaz mediante la consola o la API de OpenSearch sin servidor. Después de crear el punto de conexión de interfaz, tendrá que añadir la política del punto

de conexión al punto de conexión. Para obtener más información, consulte Acceder a Amazon OpenSearch sin servidor utilizando un punto de conexión de interfaz (AWS PrivateLink).

> **Note**
>
> No puede definir una política de punto de conexión directamente en la consola de OpenSearch Service.

Una política de punto de conexión no anula ni reemplaza otras políticas basadas en identidades, políticas basadas en recursos, políticas de red o políticas de acceso a datos que haya configurado. Para obtener información sobre cómo actualizar la política de puntos de conexión, consulte Controlar el acceso a puntos de conexión de VPC con políticas de punto de conexión.

De forma predeterminada, una política de punto de conexión concede acceso completo al punto de conexión de VPC.

```
{
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": "*",
            "Action": "*",
            "Resource": "*"
        }
    ]
}
```

Si bien la política de puntos de conexión de VPC predeterminada concede acceso total a los puntos de conexión, puede configurar una política de puntos de conexión de VPC para permitir el acceso a roles y usuarios específicos. Para eso, vea el siguiente ejemplo:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
```

```
                "AWS": [
                    "123456789012",
                    "987654321098"
                ]
            },
            "Action": "*",
            "Resource": "*"
        }
    ]
}
```

Puede especificar una colección de OpenSearch sin servidor para que se incluya como elemento condicional en su política de puntos de conexión de VPC. Para eso, vea el siguiente ejemplo:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": "*",
            "Action": "*",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aoss:collection": [
                        "coll-abc"
                    ]
                }
            }
        }
    ]
}
```

El soporte para `aoss:CollectionId` está disponible.

```
Condition": {
         "StringEquals": {
               "aoss:CollectionId": "collection-id"
```

```
        }
}
```

Puede usar las identidades de SAML en la política de puntos de conexión de VPC para determinar el acceso a esos puntos. Debe usar un comodín (*) en la sección de la entidad principal de la política de puntos de conexión de VPC. Para eso, vea el siguiente ejemplo:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": "*",
            "Action": "*",
            "Resource": "*",
            "Condition": {
                "ForAnyValue:StringEquals": {
                    "saml:cn": [
                        "saml/111122223333/idp123/group/football",
                        "saml/111122223333/idp123/group/soccer",
                        "saml/111122223333/idp123/group/cricket"
                    ]
                }
            }
        }
    ]
}
```

Además, puede configurar su política de puntos de conexión para que incluya una política de entidad principal de SAML específica. Para eso, vea lo siguiente:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
```

```
            "Principal": "*",
            "Action": "*",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:PrincipalTag/Department": [
                        "Engineering"]
                    }
                }
            }
        ]
    }
```

Para obtener más información sobre el uso de la autenticación SAML con Amazon OpenSearch sin servidor, consulte Autenticación SAML para Amazon OpenSearch sin servidor.

También puede incluir usuarios de IAM y SAML en la misma política de puntos de conexión de VPC. Para eso, vea el siguiente ejemplo:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": "*",
            "Action": "*",
            "Resource": "*",
            "Condition": {
                "ForAnyValue:StringEquals": {
                    "saml:cn": [
                        "saml/111122223333/idp123/group/football",
                        "saml/111122223333/idp123/group/soccer",
                        "saml/111122223333/idp123/group/cricket"
                    ]
                }
            }
        },
        {
```

```
            "Effect": "Allow",
            "Principal": {
                "AWS": [
                    "111122223333"
                ]
            },
            "Action": "*",
            "Resource": "*"
        }
    ]
}
```

También puede acceder a una colección de Amazon OpenSearch sin servidor desde Amazon EC2 mediante puntos de conexión de interfaz VPC. Para obtener más información, consulte Acceder a una colección de OpenSearch sin servidor desde Amazon EC2 (a través de puntos de conexión VPC de interfaz).

Consideraciones

Antes de configurar un punto de conexión para OpenSearch sin servidor, tenga en cuenta lo siguiente:

- OpenSearch sin servidor permite realizar llamadas a todas las operaciones de la API de OpenSearch (no a las operaciones de la API de configuración) a través del punto de conexión de la interfaz.
- Tras crear un punto de conexión de interfaz para OpenSearch sin servidor, aún tendrá que incluirlo en las políticas de acceso a la red para que pueda acceder a las colecciones sin servidor.
- De forma predeterminada, se permite el acceso completo OpenSearch sin servidor a través del punto de conexión. Puede asociar un grupo de seguridad a las interfaces de red de los puntos de conexión para controlar el tráfico a OpenSearch sin servidor a través del punto de conexión de interfaz.
- Una sola Cuenta de AWS puede tener un máximo de 50 puntos de conexión de VPC de OpenSearch sin servidor.
- Si habilita el acceso público a Internet a la API o los paneles de control de su colección en una política de red, cualquier VPC y la Internet pública pueden acceder a su colección.
- Si está en las instalaciones y fuera de la VPC, no puede usar directamente un resolver de DNS para la resolución del punto de conexión de VPC de OpenSearch sin servidor. Si necesita acceso

a una VPN, la VPC necesita un resolver de proxy de DNS para que lo usen los clientes externos. Route 53 ofrece una opción de punto de conexión de entrada que puede usar para resolver consultas de DNS a su VPC desde su red en las instalaciones u otra VPC.

- El servicio administra la zona alojada privada que OpenSearch sin servidor crea y adjunta a la VPC, pero aparece en los recursos de Amazon Route 53 y se factura a su cuenta.
- Para otras consideraciones, consulte Consideraciones en la Guía de AWS PrivateLink.

Permisos necesarios

La autenticación VCC para OpenSearch sin servidor utiliza los siguientes permisos AWS Identity and Access Management (IAM): Puede especificar las condiciones de IAM para restringir a los usuarios a colecciones específicas.

- `aoss:CreateVpcEndpoint`: cree un punto de conexión de VPC.
- `aoss:ListVpcEndpoints`: enumere todos los puntos de conexión de VPC.
- `aoss:BatchGetVpcEndpoint`: consulte los detalles sobre un subconjunto de puntos de conexión de VPC.
- `aoss:UpdateVpcEndpoint`: modifique un punto de conexión de VPC.
- `aoss:DeleteVpcEndpoint`: elimine un punto de conexión de VPC.

Además, necesita los siguientes permisos de Amazon EC2 y Route 53 para crear un punto de conexión de VPC.

- `ec2:CreateTags`
- `ec2:CreateVpcEndpoint`
- `ec2:DeleteVpcEndPoints`
- `ec2:DescribeSecurityGroups`
- `ec2:DescribeSubnets`
- `ec2:DescribeVpcEndpoints`
- `ec2:DescribeVpcs`
- `ec2:ModifyVpcEndPoint`
- `route53:AssociateVPCWithHostedZone`
- `route53:ChangeResourceRecordSets`

- `route53:CreateHostedZone`
- `route53:DeleteHostedZone`
- `route53:GetChange`
- `route53:GetHostedZone`
- `route53:ListHostedZonesByName`
- `route53:ListHostedZonesByVPC`
- `route53:ListResourceRecordSets`

Creación de un punto de conexión de interfaz para OpenSearch sin servidor

Puede crear un punto de conexión de interfaz para OpenSearch sin servidor mediante la consola o la API de OpenSearch sin servidor.

Cómo crear un punto de conexión de interfaz para OpenSearch sin servidor

1. Abra la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. En el panel de navegación de la izquierda, expanda Sin servidor y seleccione Puntos de conexión de VPC.
3. Seleccione Crear punto de conexión de VPC.
4. Proporcione un nombre para el punto de conexión.
5. Para VPC, seleccione la VPC desde la que accederá a OpenSearch sin servidor.
6. En Subredes, seleccione la subred desde la que accederá a OpenSearch sin servidor.
    - La dirección IP y el tipo de DNS del punto de conexión se basan en el tipo de subred.
        - Dualstack: si todas las subredes tienen rangos de direcciones IPv4 y IPv6
        - IPv6: si todas las subredes son subredes IPv6
        - IPv4: si todas las subredes tienen rangos de direcciones IPv4
7. En Grupos de seguridad, seleccione los grupos de seguridad que deban asociarse a las interfaces de red del punto de conexión. Este es un paso fundamental en el que limita los puertos, protocolos y orígenes para el tráfico entrante que autoriza para el punto de conexión. Asegúrese de que las reglas del grupo de seguridad permitan a los recursos que utilizarán los puntos de conexión de VPC comunicarse con OpenSearch sin servidor para, a su vez, comunicarse con la interfaz de la red del punto de conexión.
8. Seleccione Crear punto de conexión.

Para crear un punto de conexión de VPC mediante la API de OpenSearch sin servidor, utilice el comando `CreateVpcEndpoint`.

> **Note**
>
> Después de crear un punto de conexión, registre su ID (por ejemplo, `vpce-abc123def4EXAMPLE`. Para proporcionar el acceso del punto de conexión a sus colecciones, debe incluir este ID en una o más políticas de acceso a la red.

Tras crear un punto de conexión de interfaz, debe proporcionarles acceso a las colecciones mediante políticas de acceso a la red. Para obtener más información, consulte the section called "Acceso a la red".

Configuración de VPC compartida para Amazon OpenSearch sin servidor

Usted puede utilizar la nube privada virtual (VPC) de Amazon para compartir subredes de VPC con otras Cuentas de AWS de su organización, así como compartir infraestructura de redes, como una VPN, entre recursos en varias Cuentas de AWS.

Actualmente, Amazon OpenSearch sin servidor no admite la creación de una conexión AWS PrivateLink en una VPC compartida a menos que usted sea el propietario de esa VPC. AWS PrivateLink tampoco admite compartir conexiones entre Cuentas de AWS.

Sin embargo, gracias a la arquitectura flexible y modular de OpenSearch sin servidor, aún puede configurar una VPC compartida. Esto es posible porque la infraestructura de redes de OpenSearch sin servidor es independiente de la infraestructura individual de cada colección (OpenSearch Service). Por lo tanto, usted puede crear un punto de conexión VPCe de AWS PrivateLink en una cuenta donde se encuentre ubicada una VPC y, luego, usar el ID de ese VPCe en la política de redes de otras cuentas para restringir el tráfico, de modo que solo provenga de esa VPC compartida.

Los siguientes procedimientos se refieren a una cuenta propietaria y una cuenta consumidora.

Un propietario de la cuenta actúa como una cuenta de redes común en la que se configura una VPC y se comparte con otras cuentas. Las cuentas consumidoras son aquellas cuentas que crean y mantienen sus colecciones de OpenSearch Serverless dentro de la VPC que el propietario comparte con ellas.

Requisitos previos

Asegúrese de que se cumplan los siguientes requisitos antes de configurar la VPC compartida:

- El propietario de la cuenta debe haber configurado previamente una VPC, subredes, tabla de enrutamiento y otros recursos necesarios en nube privada virtual de Amazon. Para obtener más información, consulte la Guía del usuario de Amazon VPC.
- El propietario de la cuenta y las cuentas consumidoras deben pertenecer a la misma organización en AWS Organizations. Para obtener más información, consulte la Guía del usuario de AWS Organizations.

Para configurar una VPC compartida en una cuenta propietaria o cuenta de redes común:

1. Inicie sesión en la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. Siga los pasos de the section called “Crear un punto de conexión de interfaz”. Mientras realiza el procedimiento, haga las siguientes selecciones:
   - Seleccione una VPC y subredes que estén compartidas con las cuentas consumidoras de su organización.
3. Después de crear el punto de conexión, tome nota del ID de VPCe generado y proporciónelo a los administradores que deban realizar la tarea de configuración en las cuentas consumidoras.

   Los ID de VPCe tienen el formato `vpce-abc123def4EXAMPLE`.

Para configurar una VPC compartida en una cuenta consumidora:

1. Inicie sesión en la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. Use la información de the section called “Administración de colecciones” para crear una colección, si aún no tiene una.
3. Use la información de the section called “Creación de políticas de (consola)” para crear una política de red. A medida que avanza, realice las siguientes selecciones.

   > **Note**
   >
   > También puede actualizar una política de red existente para este fin.

   a. En Tipo de acceso, seleccione VPC (recomendado).

b. Para acceder a los puntos de conexión de VPC, elija el ID de vPCE que le proporcionó la cuenta propietaria, en el formato `vpce-abc123def4EXAMPLE`.

c. En el área Tipo de recurso, haga lo siguiente:

- Seleccione la casilla Habilitar el acceso al punto de conexión de OpenSearch y, a continuación, seleccione el nombre o el patrón de la colección que se utilizará para habilitar el acceso desde esa VPC compartida.
- Seleccione la casilla Habilitar acceso a OpenSearch Dashboard y luego seleccione el nombre de la colección o el patrón de colección que desee utilizar para habilitar el acceso desde esa VPC compartida.

4. Para una nueva política, seleccione Crear. Para una política existente, seleccione Actualizar.

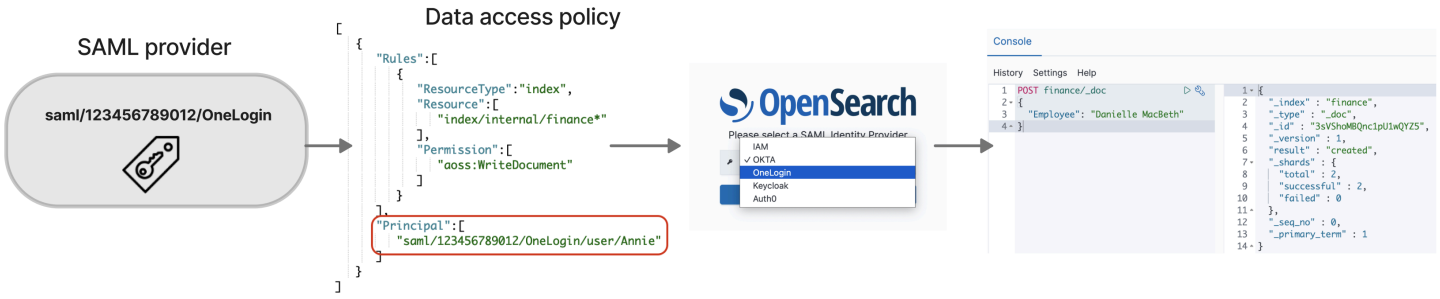
Autenticación SAML para Amazon OpenSearch sin servidor

Con la autenticación SAML para Amazon OpenSearch sin servidor, puede utilizar su proveedor de identidades existente con el fin de brindar inicio de sesión único (SSO) para OpenSearch Dashboards de colecciones sin servidor.

La autenticación SAML le permite utilizar proveedores de identidad de terceros para iniciar sesión en OpenSearch Dashboards a fin de indexar y buscar datos. OpenSearch sin servidor admite proveedores que utilizan el estándar SAML 2.0, como IAM Identity Center, Okta, Keycloak, Active Directory Federation Services (AD FS) y Auth0. Puede configurar IAM Identity Center para sincronizar usuarios y grupos de otros orígenes de identidades, como Okta, OneLogin y Microsoft Entra ID. Para obtener una lista de los orígenes de identidades que admite IAM Identity Center y los pasos para configurarlos, consulte Getting started tutorials en la Guía del usuario de IAM Identity Center.

> **Note**
>
> La autenticación SAML es solo para acceder a OpenSearch Dashboards a través de un navegador web. Los usuarios autenticados solo pueden realizar solicitudes a las operaciones de la API de OpenSearch a través de las herramientas de desarrollo de OpenSearch Dashboards. Las credenciales de SAML no permiten realizar solicitudes HTTP directas a las operaciones de la API de OpenSearch.

Para configurar la autenticación SAML, primero debe configurar un proveedor de identidades (IdP) SAML. A continuación, se incluyen uno o más usuarios de ese IdP en una política de acceso a datos. Esta política le otorga ciertos permisos para las colecciones o los índices. A continuación, el usuario puede iniciar sesión en OpenSearch Dashboards y realizar las acciones permitidas en la política de acceso a los datos.



Temas

- Consideraciones
- Permisos necesarios
- Crear proveedores SAML de (consola)
- Acceso a OpenSearch Dashboards
- Concesión de acceso a las identidades de SAML a los datos de la colección
- Creación de proveedores SAML (AWS CLI)
- Ver proveedores SAML
- Actualización de proveedores SAML
- Eliminar proveedores SAML

Consideraciones

A la hora de configurar la autenticación SAML, tenga en cuenta lo siguiente:

- No se admiten las solicitudes firmadas y cifradas.
- No se admiten las aserciones cifradas.
- No se admite la autenticación ni el cierre de sesión iniciada por el IDP.
- Las políticas de control de servicios (SCP) no se aplicarán ni evaluarán en el caso de las identidades que no sean de IAM (como SAML en Amazon OpenSearch sin servidor y SAML y la autorización de usuario interna básica para Amazon OpenSearch Service).

Permisos necesarios

La autenticación SAML para OpenSearch sin servidor utiliza los siguientes permisos AWS Identity and Access Management (IAM):

- `aoss:CreateSecurityConfig`: cree un proveedor SAML.
- `aoss:ListSecurityConfig`: enumere todos los proveedores de SAML en la cuenta actual.
- `aoss:GetSecurityConfig`: vea la información del proveedor de SAML.
- `aoss:UpdateSecurityConfig`: modifique la configuración de proveedor de SAML determinada, incluidos los metadatos XML.
- `aoss:DeleteSecurityConfig`: elimine un proveedor SAML.

La siguiente política de acceso basada en la identidad permite al usuario administrar todas las configuraciones de IdP:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Action": [
                "aoss:CreateSecurityConfig",
                "aoss:DeleteSecurityConfig",
                "aoss:GetSecurityConfig",
                "aoss:UpdateSecurityConfig",
                "aoss:ListSecurityConfigs"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

Tenga en cuenta que el elemento `Resource` debe ser un comodín.

Crear proveedores SAML de (consola)

En estos pasos se explica cómo crear proveedores de SAML. Esto permite la autenticación SAML con autenticación iniciada por el proveedor de servicios (SP) para OpenSearch Dashboards. No se admite la autenticación IdP.

Para habilitar la autenticación SAML para OpenSearch Dashboards

1. Inicie sesión en la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. En el panel de navegación de la izquierda, expanda Sin servidor y seleccione Autenticación SAML.
3. Seleccione Agregar proveedor SAML.
4. Escriba un nombre y una descripción para el proveedor.

   > **Note**
   >
   > El nombre que especifique es de acceso público y aparecerá en un menú desplegable cuando los usuarios inicien sesión en OpenSearch Dashboards. Asegúrese de que el nombre sea fácilmente reconocible y no revele información confidencial sobre su proveedor de identidad.

5. En Configurar IdP, copie la URL del servicio al consumidor de aserciones (ACS).
6. Utilice la URL de ACS que acaba de copiar para configurar su proveedor de identidades. La terminología y los pasos varían según el proveedor. Consulte la documentación de su proveedor.

   En Okta, por ejemplo, se crea una “aplicación web SAML 2.0” y se especifica la URL de ACS como URL de inicio de sesión única, URL de destinatario y URL de destino. Para Auth0, debe especificarlo en las URL de devolución de llamadas permitidas.
7. Indique la restricción de audiencia si su IdP tiene un campo para ello. La restricción de audiencia es un valor dentro de la aserción SAML que especifica a quién va dirigida esta. Con OpenSearch sin servidor, puede hacer lo siguiente. Asegúrese de reemplazar el *contenido* del siguiente ejemplo de código por su propia ID de Cuenta de AWS:

   1. Use la restricción de audiencia predeterminada `:opensearch:`*111122223333*.

2. (Opcional) configure una restricción de audiencia personalizada mediante AWS CLI. Para obtener más información, consulte Creación de proveedores SAML (AWS CLI).

   El nombre del campo de restricción de audiencia varía según el proveedor. Para Okta, es Audiencia URI (ID de entidad del SP). En el caso de IAM Identity Center, es la audiencia de Application SAML.

8. Si utiliza IAM Identity Center, también debe especificar el siguiente asignación de atributos: `Subject=${user:name}`, con un formato de `unspecified`.
9. Después de configurar el proveedor de identidades, genera un archivo de metadatos de IdP. Este archivo XML contiene información sobre el proveedor, como un certificado TLS, puntos de conexión de inicio de sesión único y el ID de entidad del proveedor de identidad.

   Copie el texto del archivo de metadatos del IdP y péguelo en Proporcionar metadatos desde el campo de IdP. Alternativamente, seleccione Importar desde archivo XML y cargue el archivo. El archivo de metadatos debe tener un aspecto similar al siguiente:

```
<?xml version="1.0" encoding="UTF-8"?>
<md:EntityDescriptor entityID="entity-id"
 xmlns:md="urn:oasis:names:tc:SAML:2.0:metadata">
  <md:IDPSSODescriptor WantAuthnRequestsSigned="false"
 protocolSupportEnumeration="urn:oasis:names:tc:SAML:2.0:protocol">
    <md:KeyDescriptor use="signing">
      <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
        <ds:X509Data>
          <ds:X509Certificate>tls-certificate</ds:X509Certificate>
        </ds:X509Data>
      </ds:KeyInfo>s
    </md:KeyDescriptor>
    <md:NameIDFormat>urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified</
md:NameIDFormat>
    <md:NameIDFormat>urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress</
md:NameIDFormat>
    <md:SingleSignOnService Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-
POST" Location="idp-sso-url"/>
    <md:SingleSignOnService Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-
Redirect" Location="idp-sso-url"/>
  </md:IDPSSODescriptor>
</md:EntityDescriptor>
```

10. Mantenga vacío el campo Atributo ID de usuario personalizado para utilizar el elemento `NameID` de la aserción SAML para el nombre de usuario. Si su aserción no utiliza este elemento estándar y, en su lugar, incluye el nombre de usuario como un atributo personalizado, especifique ese atributo aquí. Los atributos distinguen entre mayúsculas y minúsculas. Solo se admite un único atributo de usuario.

    El siguiente ejemplo muestra un atributo de anulación para `NameID` en la aserción SAML:

    ```
    <saml2:Attribute Name="UserId" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
    format:basic">
      <saml2:AttributeValue xmlns:xs="http://www.w3.org/2001/XMLSchema"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:type="xs:string">annie</saml2:AttributeValue>
    </saml2:Attribute>
    ```

11. (Opcional) Especifique un atributo personalizado en el campo Atributos del grupo, como `role` o `group`. Solo se admite un único atributo de grupo. No hay ningún atributo de grupo predeterminado. Si no especifica ninguna, sus políticas de acceso a datos solo pueden contener entidades principales de usuario.

    El siguiente ejemplo muestra un atributo de grupo en la aserción SAML:

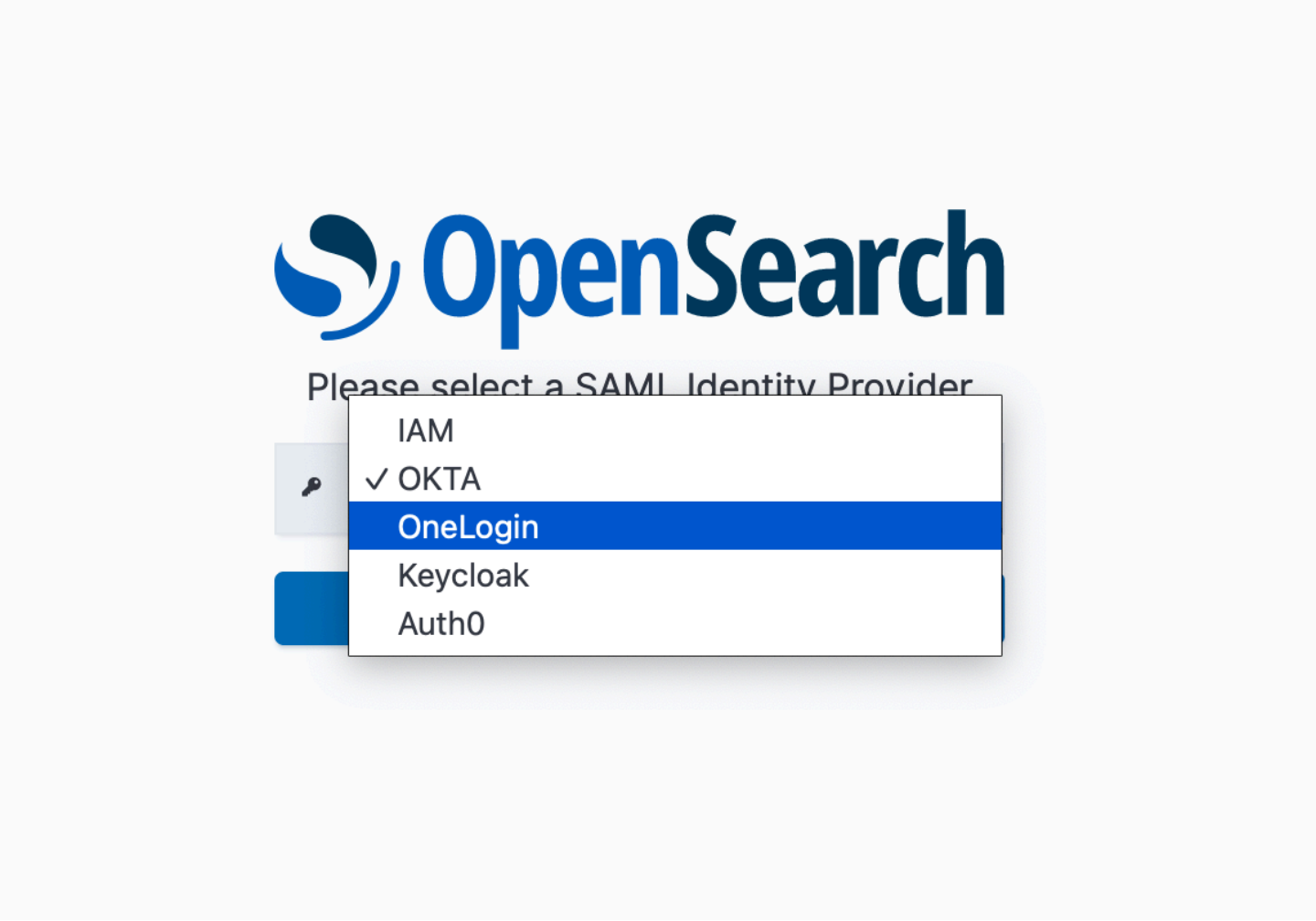
    ```
    <saml2:Attribute Name="department"
     NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:basic">
        <saml2:AttributeValue xmlns:xs="http://www.w3.org/2001/XMLSchema"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:type="xs:string">finance</saml2:AttributeValue>
    </saml2:Attribute>
    ```

12. De manera predeterminada, la sesión de los usuarios de OpenSearch Dashboards se cierra al cabo de 24 horas. Puede establecer este valor en cualquier número comprendido entre 1 y 12 horas (15 y 720 minutos) especificando el Tiempo de espera de OpenSearch Dashboards. Si intenta establecer el tiempo de espera igual o inferior a 15 minutos, la sesión se restablecerá a una hora.
13. Seleccione Crear proveedor.

Acceso a OpenSearch Dashboards

Después de configurar un proveedor de SAML, todos los usuarios y grupos asociados a ese proveedor pueden navegar hasta el punto de conexión de OpenSearch Dashboards. La URL de Dashboards tiene el formato *collection-endpoint*`/_dashboards/` para todas las colecciones.

Si tiene SAML activado, al seleccionar el enlace que aparece en la Consola de administración de AWS, será dirigido a la página de selección de IdP, donde puede iniciar sesión con sus credenciales de SAML. Primero, use el menú desplegable para seleccionar un proveedor de identidades:



Inicie sesión con sus credenciales de usuario de IdP.

Si no tiene SAML activado, al seleccionar el enlace que aparece en la Consola de administración de AWS, se le indica que inicie sesión como rol o usuario de IAM, sin opción de usar SAML.

Concesión de acceso a las identidades de SAML a los datos de la colección

Después de crear un proveedor de SAML, aún tendrá que concederles acceso a los datos de sus colecciones a los usuarios y grupos subyacentes. Para concederles el acceso deberá hacerlo a través de políticas de acceso a datos. Hasta que no les proporcione acceso a los usuarios, estos no podrán leer, escribir ni eliminar ningún dato de sus colecciones.

Para conceder el acceso, cree una política de acceso a los datos y especifique sus ID de usuario o grupo de SAML en la declaración de `Principal`:

```
[
   {
      "Rules":[
       ...
      ],
      "Principal":[
         "saml/987654321098/myprovider/user/Shaheen",
         "saml/987654321098/myprovider/group/finance"
      ]
   }
]
```

Puede concederles acceso a colecciones, índices o ambos. Si desea que diferentes usuarios tengan permisos diferentes, cree varias reglas. Para obtener una lista de los permisos disponibles, consulte Permisos de políticas compatibles. Para obtener información sobre cómo dar formato a una política de acceso, consulte Sintaxis de políticas.

Creación de proveedores SAML (AWS CLI)

Para crear un proveedor de SAML mediante la API de OpenSearch sin servidor, envíe una solicitud CreateSecurityConfig:

```
aws opensearchserverless create-security-config \
    --name myprovider \
    --type saml \
    --saml-options file://saml-auth0.json
```

Especifique `saml-options`, incluido el XML de los metadatos, como un mapa clave-valor dentro de un archivo .json. El XML de los metadatos debe estar codificado como una cadena de escape JSON.

```
{
```

```
   "sessionTimeout": 70,
   "groupAttribute": "department",
   "userAttribute": "userid",
   "openSearchServerlessEntityId": "aws:opensearch:111122223333:app1",
   "metadata": "EntityDescriptor xmlns=\"urn:oasis:names:tc:SAML:2.0:metadata
\" ... ... ... IDPSSODescriptor\r\n\/EntityDescriptor"
}
```

> **Note**
>
> (Opcional) configure una restricción de audiencia personalizada mediante AWS CLI. Para obtener más información, consulte Creación de proveedores SAML (AWS CLI).

Ver proveedores SAML

La siguiente solicitud ListSecurityConfigs muestra todos los proveedores de SAML de su cuenta:

```
aws opensearchserverless list-security-configs --type saml
```

La solicitud devuelve información sobre todos los proveedores de SAML existentes, incluidos los metadatos completos del IdP que genera su proveedor de identidades:

```
{
   "securityConfigDetails": [
      {
         "configVersion": "MTY2NDA1MjY4NDQ5M18x",
         "createdDate": 1664054180858,
         "description": "Example SAML provider",
         "id": "saml/111122223333/myprovider",
         "lastModifiedDate": 1664054180858,
         "samlOptions": {
            "groupAttribute": "department",
            "metadata": "EntityDescriptorxmlns=\"urn:oasis:names:tc:SAML:2.0:metadata
\" ...... ...IDPSSODescriptor\r\n/EntityDescriptor",
            "sessionTimeout": 120,
            "openSearchServerlessEntityId": "aws:opensearch:111122223333:app1",
            "userAttribute": "userid"
         }
      }
   ]
```

```
}
```

Para ver los detalles de un proveedor específico, incluida la `configVersion` para actualizaciones futuras, envíe una solicitud `GetSecurityConfig`.

Actualización de proveedores SAML

Para actualizar un proveedor de SAML mediante la consola de OpenSearch sin servidor, seleccione Autenticación SAML, seleccione su proveedor de identidad y elija Editar. Puede modificar todos los campos, incluidos los metadatos y los atributos personalizados.

Para actualizar un proveedor a través de la API de OpenSearch sin servidor, envíe una solicitud UpdateSecurityConfig e incluya el identificador de la política que se va a actualizar. También debe incluir una versión de configuración, que puede recuperar mediante los comandos `ListSecurityConfigs` o `GetSecurityConfig`. Incluir la versión más reciente garantiza que no se anule inadvertidamente un cambio realizado por otra persona.

La siguiente solicitud actualiza las opciones de SAML de un proveedor:

```
aws opensearchserverless update-security-config \
    --id saml/123456789012/myprovider \
    --type saml \
    --saml-options file://saml-auth0.json \
    --config-version MTY2NDA1MjY4NDQ5M18x
```

Especifique las opciones de configuración de SAML como un mapa clave-valor dentro de un archivo .json.

> **Important**
>
> Las actualizaciones de las opciones de SAML no son graduales. Si al realizar una actualización no especifica el valor para un parámetro en el objeto `SAMLOptions`, los valores existentes se sustituirán por valores vacíos. Por ejemplo, si la configuración actual contiene un valor para `userAttribute` y luego hace una actualización y no incluye este valor, el valor se elimina de la configuración. Asegúrese de saber cuáles son los valores existentes antes de realizar una actualización llamando a la operación `GetSecurityConfig`.

Eliminar proveedores SAML

Al eliminar un proveedor de SAML, las referencias de los usuarios y grupos asociados en las políticas de acceso a datos dejan de funcionar. Para evitar confusiones, le recomendamos que elimine todas las referencias al punto de conexión en sus políticas de acceso antes de eliminar el punto de conexión.

Para eliminar un proveedor de SAML mediante la consola de OpenSearch sin servidor, seleccione Autenticación, seleccione el proveedor y elija Eliminar.

Para eliminar un proveedor a través de la API de OpenSearch sin servidor, envíe una solicitud DeleteSecurityConfig:

```
aws opensearchserverless delete-security-config --id saml/123456789012/myprovider
```

Validación de conformidad para Amazon OpenSearch sin servidor

Auditores externos evalúan la seguridad y la conformidad de Amazon OpenSearch sin servidor como parte de varios programas de conformidad de AWS. Entre estos, se incluyen SOC, PCI e HIPAA.

Para saber si un Servicio de AWS está incluido en el ámbito de programas de conformidad específicos, consulte Servicios de AWS en el ámbito del programa de conformidad y escoja el programa de conformidad que le interese. Para obtener información general, consulte Programas de conformidad de AWS.

Puedes descargar los informes de auditoría de terceros utilizando AWS Artifact. Para obtener más información, consulte Descarga de informes en AWS Artifact.

Su responsabilidad de conformidad al utilizar Servicios de AWS se determina en función de la confidencialidad de los datos, los objetivos de conformidad de su empresa, así como de la legislación y los reglamentos aplicables. Para obtener más información sobre la responsabilidad de conformidad al usar Servicios de AWS, consulte la Documentación de seguridad de AWS.

Etiquetado de colecciones de Amazon OpenSearch sin servidor

Las etiquetas permiten asignar información arbitraria a un dominio de Amazon OpenSearch sin servidor para poder categorizar y filtrar esa información. Una etiqueta es un elemento de metadatos que usted o AWS asigna a un recurso de AWS.

Cada etiqueta consta de una clave y un valor. En el caso de etiquetas que usted asigna, debe definir la clave y el valor. Por ejemplo, puede definir la clave como `stage` y el valor de un recurso como `test`.

Con las etiquetas, puede identificar y organizar los recursos de AWS. Muchos servicios de AWS admiten el etiquetado, por lo que puede asignar la misma etiqueta a los recursos de diferentes servicios para indicar que los recursos están relacionados. Por ejemplo, podría asignarle la misma etiqueta a una colección de OpenSearch sin servidor que se le concede a un dominio de Amazon OpenSearch Service.

En OpenSearch sin servidor, el recurso principal es una colección. Puede utilizar la consola de OpenSearch Service, la AWS CLI, las operaciones de la API de OpenSearch sin servidor o los AWS SDK para agregar, administrar y eliminar etiquetas de una colección.

Permisos necesarios

OpenSearch sin servidor utiliza los siguientes permisos AWS Identity and Access Management Access Analyzer (IAM) para etiquetar colecciones:

- `aoss:TagResource`
- `aoss:ListTagsForResource`
- `aoss:UntagResource`

Etiquetado de colecciones (consola)

La consola es la forma más sencilla para etiquetar un dominio.

Para crear una etiqueta (consola)

1. Inicie sesión en la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. Expanda Serverless (Sin servidor) en el panel de navegación de la izquierda y seleccione Collections (Colecciones).
3. Seleccione la colección a la que desee agregar etiquetas y vaya a la pestaña Tags (Etiquetas).
4. Seleccione Administrar y Add new tag (Agregar nueva etiqueta).
5. Introduzca una clave de etiqueta y un valor opcional.

6. Seleccione guardar.

Para eliminar una etiqueta, siga los mismos pasos y elija Quitar en la página Administrar etiquetas.

Para obtener más información sobre cómo utilizar la consola para trabajar con etiquetas, consulte Tag Editor en la Guía de introducción a la consola de administración de AWS.

Etiquetado de colecciones (AWS CLI)

Para etiquetar una colección mediante la AWS CLI, envíe una solicitud de TagResource:

```
aws opensearchserverless tag-resource
  --resource-arn arn:aws:aoss:us-east-1:123456789012:collection/my-collection
  --tags Key=service,Value=aoss Key=source,Value=logs
```

Consulte las etiquetas existentes para una colección con el comando ListTagsForResource:

```
aws opensearchserverless list-tags-for-resource
  --resource-arn arn:aws:aoss:us-east-1:123456789012:collection/my-collection
```

Elimine las etiquetas de una colección mediante el comando UntagResource:

```
aws opensearchserverless untag-resource
  --resource-arn arn:aws:aoss:us-east-1:123456789012:collection/my-collection
  --tag-keys service
```

Operaciones y complementos compatibles en Amazon OpenSearch sin servidor

Amazon OpenSearch sin servidor admite una variedad de complementos de OpenSearch, así como un subconjunto de las operaciones de API de indexación, búsqueda y metadatos disponibles en esta. Puede incluir los permisos en la columna izquierda de la tabla dentro de las políticas de acceso a datos para limitar el acceso a determinadas operaciones.

Temas

- Permisos y operaciones de la API de OpenSearch
- Complementos de OpenSearch compatibles

Permisos y operaciones de la API de OpenSearch

La siguiente tabla muestra las operaciones de API que admite OpenSearch sin servidor, junto con sus correspondientes permisos de política de acceso a los datos:

Permiso de política de acceso a datos	Operaciones de la API de OpenSearch	Descripción y advertencias
`aoss:CreateIndex`	PUT <index>	Crea índices. Para obtener más información, consulte la sección Crear un índice. **Note** Este permiso también se aplica a la creación de índices con los datos de muestra en OpenSearch Dashboards.
`aoss:DescribeIndex`	• GET <index> • GET <index>/_mapping • GET <index>/_mappings • GET <index>/_setting • GET <index>/_setting/ <setting> • GET <index>/_settings • GET <index>/_settings/ <setting> • GET _cat/indices • GET _mapping • GET _mappings • GET _resolve/index/ <index> • HEAD <index>	Describe los índices. Para obtener más información, consulte los siguientes recursos: • Obtener índice • Obtener un mapeo • Obtener la configuración • Index exists • CAT indices (la respuesta no incluye los campos `health` ni `status`).

Permiso de política de acceso a datos	Operaciones de la API de OpenSearch	Descripción y advertencias
`aoss:WriteDocument`	• DELETE <index>/_doc/<id> • POST <index>/_bulk • POST <index>/_doc/ <id> (solo para tipos de colecciones de búsquedas) • POST <index>/_doc • POST <index>/_update/<id> (solo para tipos de colecciones de búsquedas) • POST _bulk • PUT <index>/_create/<id> (solo para tipos de colecciones de búsquedas) • PUT <index>/_doc/<id> (solo para tipos de colecciones de búsquedas)	Redacta y actualiza documentos. Para obtener más información, consulte los recursos de siguientes: • Bulk • Índice de datos **Note** Algunas operaciones solo están permitidas para colecciones de tipos `SEARCH`. Para obtener más información, consulte the section called “Elección de un tipo de colección”.

Permiso de política de acceso a datos	Operaciones de la API de OpenSearch	Descripción y advertencias
`aoss:ReadDocument`	• GET <index>/_analyz • GET <index>/_doc/ <id> • GET <index>/_explain/ <id> • GET <index>/_mget • GET <index>/_source/ <id> • GET <index>/_count • GET <index>/_field_caps • GET <index>/_msearch • GET <index>/_rank_eval • GET <index>/_search • GET <index>/_validate/ <query> • GET _analyze • GET _field_caps • GET _mget • GET _search • GET /_search/point_in_time/_all • HEAD <index>/_doc/ <id> • HEAD <index>/_source/ <id> • POST /_plugins/_sql • POST /_plugins/_ppl • POST /_plugins/_sql/_explain • POST /_plugins/_ppl/_explain • POST /_plugins/_ppl/_close • POST <index>/_analyze • POST /_search/point_in_time • POST <index>/_explain/ <id> • POST <index>/_count • POST <index>/_field_caps	Lee los documentos. Para obtener más información, consulte los siguientes recursos: • Realice el análisis de texto • Obtener el documento • recuento • Consulte DSL • Evaluación de clasificación • Analice la API • Explain • Momento dado • SQL y PPL

Permiso de política de acceso a datos	Operaciones de la API de OpenSearch	Descripción y advertencias
	• POST <index>/_rank_eval • POST <index>/_search • POST _analyze • POST _field_caps • POST _search • DELETE /_search/point_in_time/_all • DELETE /_search/point_in_time	
`aoss:DeleteIndex`	DELETE <target>	Elimina índices. Para obtener más información, consulte Eliminar índices.
`aoss:UpdateIndex`	• POST _mapping • POST <index>/_mapping/ • POST <index>/_mappings/ • POST <index>/_setting • POST <index>/_setting • POST _setting • POST _settings • PUT _mapping • PUT <index>/_mapping • PUT <index>/_mapping • PUT <index>/_setting • PUT <index>/_settings • PUT _setting • PUT _settings	Actualiza las configuraciones del índice. Para obtener más información, consulte los siguientes recursos: • Correspondencia • Actualizar las configuraciones

Permiso de política de acceso a datos	Operaciones de la API de OpenSearch	Descripción y advertencias
`aoss:CreateCollectionItems`	• POST _aliases • POST /_plugins/_flow_framework/workflow • * POST /_plugins/_flow_framework/workflow/<workflow_id>/_provision • PUT _ingest/pipeline/<pipeline-id> • PUT _search/pipeline/<pipeline-id>	• Crear alias, canalizaciones y plantillas de índices. Para obtener más información, consulte Crear alias. • * Aprovisionar o volver a aprovisionar plantillas. El Cliente ML Commons y los servicios de OpenSearch sin servidor administran las políticas dependientes.

Permiso de política de acceso a datos	Operaciones de la API de OpenSearch	Descripción y advertencias
`aoss:DescribeCollectionItems`	• GET <index>/_alias/ <alias> • GET _alias • GET _alias/ <alias> • GET _cat/alias • GET _cat/templates • GET _cat/templates/ <template_name> • GET _component_template • GET _component_template/ <component-template> • GET _index_template • GET _index_template/ <index-template> • GET _ingest/pipeline/<pipeline-id> • GET _ingest/pipeline/_simulate • GET /_plugins/_flow_framework/workflow/<workflow-id> • GET /_plugins/_flow_framework/workflow/_search • GET /_plugins/_flow_framework/workflow/<workflow-id>/_status • GET /_plugins/_flow_framework/workflow/state/_search • GET /_plugins/_flow_framework/workflow/_steps • GET /_plugins/_flow_framework/workflow/_step?workflow_step=<step_name> • GET _search/pipeline/<pipeline-id>	Describe cómo trabajar con alias, plantillas de índices y marcos y canalizaciones. Para obtener más información, consulte los siguientes recursos: • Administrar alias • Plantillas del índice

Permiso de política de acceso a datos	Operaciones de la API de OpenSearch	Descripción y advertencias
	• HEAD _alias/ <alias> • HEAD _component_template/ <component-template> • HEAD _index_template/ <name> • HEAD <index>/_alias/ <alias> • POST _ingest/pipeline/_simulate • POST /_plugins/_flow_framework/workflow/_search • POST /_plugins/_flow_framework/workflow/state/_search	
`aoss:UpdateCollectionItems`	• POST <index>/_alias/ <alias> • POST <index>/_aliases/ <alias> • POST _component_template/ <component-template> • POST _index_template/ <index-template> • * POST /_plugins/_flow_framework/workflow/<workflow_id>/_deprovision • PUT <index>/_alias/ <alias> • PUT <index>/_aliases/ <alias> • PUT _component_template/ <component-template> • PUT _index_template/ <index-template> • PUT /_plugins/_flow_framework/workflow/<workflow_id>	Actualice los alias, las plantillas de índice y las plantillas de framework. Para obtener más información, consulte los siguientes recursos: • Alias de índice • Plantillas del índice *. La API para desaprovisionar plantillas. El Cliente ML Commons y los servicios de OpenSearch Sin Servidor administran las políticas dependientes.

Permiso de política de acceso a datos	Operaciones de la API de OpenSearch	Descripción y advertencias
`aoss:DeleteCollectionItems`	• DELETE <index>/_alias/ <alias> • DELETE _component_template/ <component-template> • DELETE _index_template/ <index-template> • DELETE <index>/_aliases/ <alias> • DELETE _search/pipeline/< pipeline-id> • DELETE _ingest/pipeline/< pipeline-id> • DELETE /_plugins/_flow_framework/workflow/<workflow_id>	Elimina alias, plantillas de índices y marcos, y canalizaciones. Para obtener más información, consulte los siguientes recursos: • Eliminar alias • Elimina una plantilla

Permiso de política de acceso a datos	Operaciones de la API de OpenSearch	Descripción y advertencias
`aoss:DescribeMLResource`	• GET /_plugins/_ml/models/<model_id> • GET /_plugins_ml/models/_search • GET /_plugins/_ml/model_groups/<model_group_id> • GET /_plugins/_ml/model_groups/_search • GET /_plugins/_ml/connectors/<connector_id> • GET /_plugins/_ml/connectors/_search • GET /_plugins/_ml/profile/tasks/<task_id> • POST /_plugins/_ml/models/_search • POST /_plugins/_ml/model_groups/_search • POST /_plugins/_ml/connectors/_search	Describe las API GET y de búsqueda para recuperar información sobre modelos y conectores.
`aoss:CreateMLResource`	• POST /_plugins/_ml/models/_register • POST /_plugins/_ml/model_groups/_register • POST /_plugins/_ml/connectors/_create	Proporciona permiso para crear recursos de ML.

Permiso de política de acceso a datos	Operaciones de la API de OpenSearch	Descripción y advertencias
`aoss:UpdateMLResource`	• PUT /_plugins/_ml/models/<model_id> • POST /_plugins/_ml/models/<model_id>/_deploy • POST /_plugins/_ml/models/<model_id>/_undeploy • PUT /_plugins/_ml/model_groups/<model_group_id> • PUT /_plugins/_ml/connectors/<connector_id>	Otorga permiso para actualizar recursos de aprendizaje automático existentes.
`aoss:DeleteMLResource`	• DELETE /_plugins/_ml/models/<model_id> • DELETE _plugins/_ml/model_groups/<model_group_id> • DELETE /_plugins/_ml/connectors/<connector_id> • DELETE /_plugins/_ml/tasks/<task_id>	Proporciona permiso para eliminar recursos de aprendizaje automático.
`aoss:ExecuteMLResource`	• POST /_plugins/_ml/models/<model_id>/_predict	Proporciona permiso para ejecutar modelos.

Complementos de OpenSearch compatibles

OpenSearch sin servidor viene empaquetados con complementos de la comunidad de OpenSearch. La ejecución sin servidor implementa y administra automáticamente los complementos.

Complementos de análisis

- ICU Analysis
- Japanese (kuromoji) Analysis

- Análisis coreano (Nori
- Phonetic Analysis
- Smart Chinese Analysis
- Stempel Polish Analysis
- Ukrainian Analysis

Complementos de Mapper

- Mapper Size
- Mapper Murmur3
- Texto anotado de Mapper

Complementos de la creación de scripts

- Painless
- Expression
- Mustache

Además, OpenSearch sin servidor incluye todos los complementos que se envían como módulos.

Monitoreo de Amazon OpenSearch sin servidor

El monitoreo es una parte importante del mantenimiento de la fiabilidad, la disponibilidad y el rendimiento de Amazon OpenSearch sin servidor y de sus otras soluciones de AWS. AWS ofrece las siguientes herramientas de monitoreo para vigilar a OpenSearch sin servidor, informar cuando algo no va bien y tomar medidas automáticamente cuando proceda:

- Amazon CloudWatch monitorea los recursos de AWS y las aplicaciones que ejecuta en AWS en tiempo real. Puede recopilar métricas y realizar un seguimiento de las métricas, crear paneles personalizados y definir alarmas que le advierten o que toman medidas cuando una métrica determinada alcanza el umbral que se especifique.

  Por ejemplo, puede hacer que CloudWatch haga un seguimiento del uso de la CPU u otras métricas de las instancias de Amazon EC2 y abrir nuevas instancias automáticamente cuando sea necesario. Para más información, consulte la Guía del usuario de Amazon CloudWatch.

- AWS CloudTrail captura las llamadas a la API y los eventos relacionados realizados por su cuenta de Cuenta de AWS o en su nombre. Entrega los archivos de registro al bucket de Amazon S3 que se especifique. También puede identificar qué usuarios y cuentas llamaron a AWS, la dirección IP de origen de las llamadas y el momento en que se hicieron. Para más información, consulte la Guía del usuario de AWS CloudTrail.
- Amazon EventBridge ofrece un flujo casi en tiempo real de eventos del sistema que describen los cambios en sus dominios de OpenSearch Service. Puede crear reglas que vigilen determinados eventos y desencadenen acciones automatizadas en otros Servicios de AWS cuando estos eventos se produzcan. Para más información, consulte la Guía del usuario de Amazon EventBridge.

Monitoreo de registros de OpenSearch sin servidor con Amazon CloudWatch

Puede monitorear Amazon OpenSearch sin servidor mediante CloudWatch, que recopila datos sin formato y los procesa en métricas legibles y casi en tiempo real. Estas estadísticas se mantienen durante 15 meses, de forma que pueda obtener acceso a información histórica y disponer de una mejor perspectiva sobre el rendimiento de su aplicación web o servicio.

También puede establecer alarmas que vigilen determinados umbrales y enviar notificaciones o realizar acciones cuando se cumplan dichos umbrales. Para más información, consulte la Guía del usuario de Amazon CloudWatch.

OpenSearch sin servidor informa de las siguientes métricas en el espacio de nombres `AWS/AOSS`.

Métrica	Descripción
`ActiveCollection`	Indica si una colección está activa. Un valor de 1 significa que la colección está en estado `ACTIVE`. Este valor se emite al crear correctamente una colección y permanece como 1 hasta que esta se elimine. La métrica no puede tener un valor de 0. Estadísticas pertinentes: máximo Dimensiones: `ClientId`, `CollectionId` , `CollectionName`

Métrica	Descripción
	Frecuencia: 60 segundos
`DeletedDocuments`	El número total de documentos eliminados. Estadísticas pertinentes: promedio, suma Dimensiones: `ClientId`, `CollectionId` , `CollectionName` , `IndexId`, `IndexName` Frecuencia: 60 segundos
`IndexingOCU`	La cantidad de unidades de computación de OpenSearch (OCU) utilizadas para incorporar datos de recopilación. Esta métrica se aplica según el nivel de la cuenta. Estadísticas pertinentes: suma Dimensiones: `ClientId` Frecuencia: 60 segundos
`IngestionDataRate`	La velocidad de indexación en GiB por segundo de una colección o índice. Esta métrica se aplica únicamente a las solicitudes de indexación masiva. Estadísticas pertinentes: suma Dimensiones: `ClientId`, `CollectionId` , `CollectionName` , `IndexId`, `IndexName` Frecuencia: 60 segundos

Métrica	Descripción
`IngestionDocumentErrors`	Número total de errores del documento durante el procesamiento de una colección o índice. Tras una solicitud de indexación masiva satisfactoria, los redactores procesan la solicitud y emiten errores para todos los documentos fallidos incluidos en la solicitud. Estadísticas pertinentes: suma Dimensiones: `ClientId`, `CollectionId` , `CollectionName` , `IndexId`, `IndexName` Frecuencia: 60 segundos
`IngestionDocumentRate`	La velocidad por segundo a la que se incorporan los documentos a una colección o índice. Esta métrica se aplica únicamente a las solicitudes de indexación masiva. Estadísticas pertinentes: suma Dimensiones: `ClientId`, `CollectionId` , `CollectionName` , `IndexId`, `IndexName` Frecuencia: 60 segundos
`IngestionRequestErrors`	El número total de errores de solicitud de indexación masiva en una colección. OpenSearch sin servidor emite esta métrica cuando se produce un error en una solicitud de indexación masiva por cualquier motivo, como un problema de autenticación o disponibilidad. Estadísticas pertinentes: suma Dimensiones: `ClientId`, `CollectionId` , `CollectionName` Frecuencia: 60 segundos

Métrica	Descripción
`IngestionRequestLatency`	La latencia, en segundos, de las operaciones de escritura masiva en una colección. Estadísticas pertinentes: mínimo, máximo, promedio Dimensiones: `ClientId`, `CollectionId` , `CollectionName` Frecuencia: 60 segundos
`IngestionRequestRate`	El número total de operaciones de escritura masiva recibidas por una colección. Estadísticas pertinentes: mínimo, máximo, promedio Dimensiones: `ClientId`, `CollectionId` , `CollectionName` Frecuencia: 60 segundos
`IngestionRequestSuccess`	El número de operaciones de indexación de datos de una colección. Estadísticas pertinentes: suma Dimensiones: `ClientId`, `CollectionId` , `CollectionName` Frecuencia: 60 segundos
`SearchableDocuments`	Número total de documentos susceptibles de búsqueda en una colección o índice. Estadísticas pertinentes: Suma Dimensiones: `ClientId`, `CollectionId` , `CollectionName` , `IndexId`, `IndexName` Frecuencia: 60 segundos

Métrica	Descripción
`SearchRequestErrors`	El número total de errores de consulta por minuto de una colección. Estadísticas pertinentes: suma Dimensiones: `ClientId`, `CollectionId` , `CollectionName` Frecuencia: 60 segundos
`SearchRequestLatency`	El tiempo promedio, en milisegundos, que se tarda en completar una operación de búsqueda en una colección. Estadísticas pertinentes: mínimo, máximo, promedio Dimensiones: `ClientId`, `CollectionId` , `CollectionName` Frecuencia: 60 segundos
`SearchOCU`	La cantidad de unidades informáticas (OCU) de OpenSearch utilizadas para buscar datos de recopilación. Esta métrica se aplica según el nivel de la cuenta. Estadísticas pertinentes: suma Dimensiones: `ClientId` Frecuencia: 60 segundos
`SearchRequestRate`	El número total de solicitudes de búsqueda por minuto para una colección. Estadísticas pertinentes: promedio, máximo, suma Dimensiones: `ClientId`, `CollectionId` , `CollectionName` Frecuencia: 60 segundos

Métrica	Descripción
`StorageUsedInS3`	La cantidad, en bytes, de almacenamiento de Amazon S3 utilizada. OpenSearch sin servidor almacena datos indexados en Amazon S3. Debe seleccionar el periodo en un minuto para obtener un valor preciso. Estadísticas pertinentes: suma Dimensiones: `ClientId`, `CollectionId` , `CollectionName` , `IndexId`, `IndexName` Frecuencia: 60 segundos
`2xx, 3xx, 4xx, 5xx`	El número de solicitudes para la colección que produjeron el código de respuesta HTTP especificado (2xx, 3xx, 4xx, 5xx). Estadísticas pertinentes: suma Dimensiones: `ClientId`, `CollectionId` , `CollectionName` Frecuencia: 60 segundos

Registro de llamadas a la API de OpenSearch sin servidor mediante AWS CloudTrail

Amazon OpenSearch sin servidor se integra con AWS CloudTrail, un servicio que proporciona un registro de las medidas adoptadas por un usuario, un rol o un servicio sin servidor AWS.

CloudTrail captura todas las llamadas de la API para OpenSearch sin servidor como eventos. Las llamadas capturadas incluyen llamadas desde la sección Serverless (Sin servidor) de la consola OpenSearch Service y llamadas de código a las operaciones de la API de OpenSearch sin servidor.

Si crea un registro de seguimiento, puede habilitar la entrega continua de eventos de CloudTrail a un bucket de Amazon S3, incluidos los eventos para OpenSearch sin servidor. Si no configura un registro de seguimiento, puede ver los eventos más recientes en la consola de CloudTrail en el Historial de eventos.

Mediante la información recopilada por CloudTrail, puede determinar la solicitud que se realizó en OpenSearch sin servidor, la dirección IP desde la cual se realizó, quién la realizó y cuándo, etc.

Para obtener más información sobre CloudTrail, consulte la Guía del usuario de AWS CloudTrail.

Información de Amazon OpenSearch sin servidor en CloudTrail

CloudTrail se habilita en su Cuenta de AWS cuando se crea la cuenta. Cuando se produce una actividad en OpenSearch sin servidor, esa actividad se registra en un evento de CloudTrail junto con otros eventos de servicio de AWS en Historial de eventos. Puede ver, buscar y descargar eventos recientes en su Cuenta de AWS. Para más información, consulte Viewing events with CloudTrail Event history.

Para mantener un registro continuo de los eventos en su Cuenta de AWS, incluidos los eventos de OpenSearch sin servidor, cree un registro de seguimiento. Un registro de seguimiento permite a CloudTrail enviar archivos de registro a un bucket de Amazon S3. De forma predeterminada, cuando se crea un registro de seguimiento en la consola, el registro de seguimiento se aplica a todas las Regiones de AWS.

El registro de seguimiento registra los eventos de todas las regiones de la partición de AWS y envía los archivos de registro al bucket de Amazon S3 especificado. También es posible configurar otros servicios de AWS para analizar en profundidad y actuar en función de los datos de eventos recopilados en los registros de CloudTrail. Para más información, consulte los siguientes temas:

- Introducción a la creación de registros de seguimiento
- Servicios e integraciones compatibles con CloudTrail
- Configuración de notificaciones de Amazon SNS para CloudTrail
- Recibir archivos de registro de CloudTrail de varias regiones y Recibir archivos de registro de CloudTrail de varias cuentas

CloudTrail registra todas las acciones de OpenSearch sin servidor que se documentan en la referencia de API de OpenSearch sin servidor. Por ejemplo, las llamadas a las acciones `CreateCollection`, `ListCollections` y `DeleteCollection` generan entradas en los archivos de registro de CloudTrail.

Cada entrada de registro o evento contiene información sobre quién generó la solicitud. La información de identidad lo ayuda a determinar:

- Si la solicitud se realizó con credenciales de usuario de AWS Identity and Access Management (IAM) o credenciales de usuario raíz.
- Si la solicitud se realizó con credenciales de seguridad temporales de un rol o fue un usuario federado.
- Si la solicitud la realizó otro servicio de AWS.

Para obtener más información, consulte Elemento userIdentity de CloudTrail.

Eventos de datos de OpenSearch sin servidor en CloudTrail

Los eventos de datos proporcionan información sobre las operaciones realizadas en un recurso o dentro de él (por ejemplo, búsquedas o indexaciones en una colección de OpenSearch sin servidor). Se denominan también operaciones del plano de datos. Los eventos de datos suelen ser actividades de gran volumen. De forma predeterminada, CloudTrail no registra eventos de datos. El Historial de eventos de CloudTrail no registra los eventos de datos.

Se aplican cargos adicionales a los eventos de datos. Para obtener más información sobre los precios de CloudTrail, consulte Precios de AWS CloudTrail.

Puede registrar eventos de datos del tipo de recurso de `AWS::AOSS::Collection` mediante la consola de CloudTrail, la AWS CLI o las operaciones de la API de CloudTrail. Para obtener más información sobre cómo registrar los eventos de datos, consulte Registro de eventos de datos con la Consola de administración de AWS y Registro de eventos de datos con la AWS Command Line Interface en la Guía del usuario de AWS CloudTrail.

Puede configurar selectores de eventos avanzados para filtrar según los campos `eventName`, `readOnly` y `resources.ARN` y así registrar solo los eventos que son importantes para usted. Para obtener más información acerca de estos campos, consulte AdvancedFieldSelector en la Referencia de la API de AWS CloudTrail.

Comprensión de las entradas de eventos de datos de OpenSearch sin servidor

En el siguiente ejemplo:

- El campo `requestParameters` contiene detalles sobre la llamada a la API realizada a la colección. Incluye la ruta base de la petición (sin parámetros de consulta).
- El campo `responseElements` incluye un código de estado que indica el resultado de su petición al modificar recursos. Este código de estado lo ayuda a verificar si los cambios se procesaron correctamente o si requieren atención.

- OpenSearch sin servidor registra eventos de datos de CloudTrail únicamente para las peticiones que completan de manera satisfactoria la autenticación de IAM.

```
        "accessKeyId": "access-key",
        "userName": "",
        "sessionContext": {
          "sessionIssuer": {
            "type": "Role",
            "principalId": "AROA123456789EXAMPLE",
            "arn": "arn:aws:iam::111122223333:role/Admin",
            "accountId": "111122223333",
            "userName": "Admin"
          },
          "attributes": {
            "creationDate": "2025-08-15T22:57:38Z",
            "mfaAuthenticated": "false"
          },
          "sourceIdentity": "",
          "ec2RoleDelivery": "",
          "assumedRoot": ""
        },
        "identityProvider": "",
        "credentialId": ""
      },
      "eventTime": "2025-08-15T22:58:00Z",
      "eventSource": "aoss.amazonaws.com",
      "eventName": "Search",
      "awsRegion": "us-east-1",
      "sourceIPAddress": "AWS Internal",
      "userAgent": "python-requests/2.32.3",
      "requestParameters": {
        "pathPrefix": "/_search"
      },
      "responseElements": null,
      "requestID": "2cfee788-EXAM-PLE1-8617-4018cEXAMPLE",
      "eventID": "48d43617-EXAM-PLE1-9d9c-f7EXAMPLE",
      "readOnly": true,
      "resources": [
        {
          "type": "AWS::AOSS::Collection",
          "ARN": "arn:aws:aoss:us-east-1:111122223333:collection/aab9texampletu45xh77"
        }
      ],
      "eventType": "AwsApiCall",
      "managementEvent": false,
      "recipientAccountId": "111122223333",
      "eventCategory": "Data"
    }
  ]
}
```

Comprensión de las entradas de eventos de administración de OpenSearch sin servidor

Un registro de seguimiento es una configuración que permite la entrega de eventos como archivos de registros en un bucket de Amazon S3 que especifique. Los archivos de registro de CloudTrail contienen una o varias entradas de registro.

Un evento representa una única solicitud desde cualquier origen. Incluye información sobre la acción solicitada, la fecha y la hora de la acción, los parámetros de la solicitud, etc. Los archivos de registro de CloudTrail no rastrean el orden en la pila de las llamadas públicas a la API, por lo que estas no aparecen en ningún orden específico.

El siguiente ejemplo muestra una entrada de registro de CloudTrail que demuestra la acción `CreateCollection`.


```
{
   "eventVersion":"1.08",
   "userIdentity":{
      "type":"AssumedRole",
      "principalId":"AIDACKCEVSQ6C2EXAMPLE",
      "arn":"arn:aws:iam::123456789012:user/test-user",
      "accountId":"123456789012",
      "accessKeyId":"access-key",
      "sessionContext":{
         "sessionIssuer":{
            "type":"Role",
            "principalId":"AIDACKCEVSQ6C2EXAMPLE",
            "arn":"arn:aws:iam::123456789012:role/Admin",
            "accountId":"123456789012",
            "userName":"Admin"
         },
         "webIdFederationData":{

         },
         "attributes":{
            "creationDate":"2022-04-08T14:11:34Z",
            "mfaAuthenticated":"false"
         }
      }
   },
   "eventTime":"2022-04-08T14:11:49Z",
   "eventSource":"aoss.amazonaws.com",
```

```
   "eventName":"CreateCollection",
   "awsRegion":"us-east-1",
   "sourceIPAddress":"AWS Internal",
   "userAgent":"aws-cli/2.1.30 Python/3.8.8 Linux/5.4.176-103.347.amzn2int.x86_64 exe/
x86_64.amzn.2 prompt/off command/aoss.create-collection",
   "errorCode":"HttpFailureException",
   "errorMessage":"An unknown error occurred",
   "requestParameters":{
      "accountId":"123456789012",
      "name":"test-collection",
      "description":"A sample collection",
      "clientToken":"d3a227d2-a2a7-49a6-8fb2-e5c8303c0718"
   },
   "responseElements": null,
   "requestID":"12345678-1234-1234-1234-987654321098",
   "eventID":"12345678-1234-1234-1234-987654321098",
   "readOnly":false,
   "eventType":"AwsApiCall",
   "managementEvent":true,
   "recipientAccountId":"123456789012",
   "eventCategory":"Management",
   "tlsDetails":{
      "clientProvidedHostHeader":"user.aoss-sample.us-east-1.amazonaws.com"
   }
}
```


Monitoreo de eventos de OpenSearch sin servidor con Amazon EventBridge

Amazon OpenSearch Service se integra con Amazon EventBridge para informar sobre determinados eventos que afectan a los dominios. Los eventos de los servicios de AWS se envían a EventBridge casi en tiempo real. Los mismos eventos también se envían a Eventos de Amazon CloudWatch, el predecesor de Amazon EventBridge. Puede crear reglas para indicar qué eventos le resultan de interés, así como qué acciones automatizadas se van a realizar cuando un evento cumple una de las reglas. Entre los ejemplos de acciones que puede activar automáticamente se incluyen los siguientes:

- Invocar una función de AWS Lambda
- Invocar un Run Command de Amazon EC2
- Desviar el evento a Amazon Kinesis Data Streams

- Activación de una máquina de estado de AWS Step Functions
- Notificar un tema de Amazon SNS o una cola de Amazon SQS

Para más información, consulte Introducción a Amazon EventBridge en la Guía del usuario de Amazon EventBridge.

Configuración de notificaciones

Puede utilizar las notificaciones de usuario de AWS para recibir notificaciones cuando se produzca un evento de OpenSearch sin servidor. Un evento es un indicador de un cambio en el entorno OpenSearch sin servidor, por ejemplo, cuando se alcanza el límite máximo de uso de OCU. Amazon EventBridge recibe el evento y envía una notificación al Centro de notificaciones de la Consola de administración de AWS y a los canales de entrega que haya elegido. Recibirá una notificación cuando un evento coincida con una regla que especifique.

Eventos de unidades de cómputo (OCU) de OpenSearch

OpenSearch sin servidor envía eventos a EventBridge cuando se produce uno de los siguientes eventos relacionados con OCU.

Uso de OCU cercano al límite máximo

OpenSearch sin servidor envía este evento cuando el uso de OCU de búsqueda o indexación alcanza el 75 % del límite de capacidad. El uso de OCU se calcula en función del límite de capacidad configurado y del consumo actual de OCU.

Ejemplo

A continuación, se muestra un ejemplo de este evento (OCU de búsqueda):

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "OCU Utilization Approaching Max Limit",
  "source": "aws.aoss",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
```

```
    "eventTime" : 1678943345789,
    "description": "Your search OCU usage is at 75% and is approaching the configured
 maximum limit."
  }
}
```

A continuación, se muestra un ejemplo de este evento (OCU de indexación):

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "OCU Utilization Approaching Max Limit",
  "source": "aws.aoss",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "eventTime" : 1678943345789,
    "description": "Your indexing OCU usage is at 75% and is approaching the configured
 maximum limit."
  }
```

El uso de OCU ha alcanzado el límite máximo

OpenSearch sin servidor envía este evento cuando el uso de OCU de búsqueda o indexación alcanza el 100 % del límite de capacidad. El uso de OCU se calcula en función del límite de capacidad configurado y del consumo actual de OCU.

Ejemplo

A continuación, se muestra un ejemplo de este evento (OCU de búsqueda):

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "OCU Utilization Reached Max Limit",
  "source": "aws.aoss",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
```

```
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "eventTime" : 1678943345789,
    "description": "Your search OCU usage has reached the configured maximum limit."
  }
}
```

A continuación, se muestra un ejemplo de este evento (OCU de indexación):

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "OCU Utilization Reached Max Limit",
  "source": "aws.aoss",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "eventTime" : 1678943345789,
    "description": "Your indexing OCU usage has reached the configured maximum limit."
  }
}
```

Creación y administración de dominios de Amazon OpenSearch Service

En este capítulo se describe cómo crear y administrar dominios de Amazon OpenSearch Service. Un dominio es el equivalente aprovisionado por AWS de un clúster de OpenSearch de código abierto. Al crear un dominio, especifica la configuración, los tipos de instancias, los recuentos de instancias y la asignación de almacenamiento. Para obtener más información sobre los clústeres de código abierto, consulte Creating a cluster en la documentación de OpenSearch.

A diferencia de las breves instrucciones incluidas en el tutorial de introducción, en este capítulo se describen todas las opciones y se proporciona información de referencia relevante. Puede completar todos los procedimientos mediante las instrucciones para la consola de OpenSearch Service, la AWS Command Line Interface (AWS CLI) o los SDK de AWS.

Creación de dominios de OpenSearch Service

Esta sección describe cómo crear dominios de OpenSearch Service desde la consola de OpenSearch Service o mediante la AWS CLI con el comando `create-domain`.

Creación de dominios de OpenSearch Service (consola)

Utilice el siguiente procedimiento para crear un dominio de OpenSearch Service mediante la consola.

Para crear un dominio de OpenSearch Service (consola)

1. Visite https://aws.amazon.com y elija Sign In to the Console (Iniciar sesión en la consola).
2. En Análisis, seleccione Amazon OpenSearch Service.
3. Seleccione Create domain (Crear un dominio).
4. Para el Nombre de dominio, escriba un nombre de dominio. El nombre debe cumplir los siguientes criterios:
   - Ser único en la cuenta y en la Región de AWS
   - Comenzar por una letra minúscula.
   - Contener entre 3 y 28 caracteres.
   - Contiene únicamente letras minúsculas a-z, números de 0-9 y un guion (-).

5. Seleccione Creación estándar para el método de creación del dominio.
6. En Plantillas, seleccione la opción que mejor se ajuste a la finalidad del dominio:

   - Dominios de producción para cargas de trabajo que necesitan alta disponibilidad y rendimiento. Estos dominios utilizan Multi-AZ (con o sin modo de espera) y nodos principales dedicados para una mayor disponibilidad.
   - Desarrollo/pruebas para desarrollo o pruebas. Estos dominios pueden usar Multi-AZ (con o sin modo de espera) o una única zona de disponibilidad.

   > **Important**
   >
   > Los diversos tipos de implementación presentarán opciones distintas en las páginas siguientes. Estos pasos incluyen todas las opciones.

7. En cuanto a las Opciones de implementación, seleccione Dominio con modo de espera para configurar un dominio 3-AZ, con los nodos de una de las zonas reservados como don modo de espera. Esta opción aplica una serie de prácticas recomendadas, como un recuento de nodos de datos específico, un recuento de nodos maestros, un tipo de instancia, un recuento de réplicas y una configuración de actualización de software.
8. En Version (Versión), elija la versión de OpenSearch o Elasticsearch OSS heredado que desea utilizar. Recomendamos elegir la última versión de OpenSearch. Para obtener más información, consulte the section called “Versiones compatibles de Elasticsearch y OpenSearch”.

   (Opcional) Si eligió una versión de OpenSearch para un dominio, seleccione Enable compatibility mode (Habilitar el modo de compatibilidad) para que OpenSearch informe que la versión es 7.10, lo cual permite a ciertos clientes y complementos de Elasticsearch OSS verificar la versión antes de conectarse a fin de continuar trabajando con el servicio.
9. En Instance type (Tipo de instancia), elija un tipo de instancia para los nodos de datos. Para obtener más información, consulte the section called “Tipos de instancias admitidas”.

   > **Note**
   >
   > No todas las zonas de disponibilidad admiten todos los tipos de instancias. Si elige Multi-AZ con o sin modo de espera, recomendamos elegir tipos de instancias de la generación actual, como R5 o I3.

10. Para el Número de nodos, elija el número de nodos de datos.

Para ver los valores máximos, consulte las Cuotas de dominios e instancias de OpenSearch Service. Los clústeres de un solo nodo son adecuados para desarrollo y pruebas, pero no deben utilizarse para cargas de trabajo de producción. Para obtener más información, consulte the section called “Determinación del tamaño de dominios” y the section called “Configuración de un dominio multi-AZ”.

> **Note**
>
> (Opcional) Los nodos coordinadores dedicados admiten todas las versiones de OpenSearch y las versiones entre la 6.8 y la 7.10 de Elasticsearch. Los nodos coordinadores dedicados están disponible para su uso con dominios que tienen un administrador de clústeres dedicado habilitado. Para habilitar los nodos coordinadores dedicados, seleccione el tipo y el recuento de instancias. Como práctica recomendada, debe mantener la familia de instancias de su nodo coordinador dedicado igual que la de sus nodos de datos (instancias basadas en Intel o instancias basadas en Graviton).

11. Para Tipo de almacenamiento, seleccione Amazon EBS. Los tipos de volumen disponibles en la lista dependen del tipo de instancia que haya elegido. Para obtener instrucciones sobre cómo crear dominios especialmente grandes, consulte the section called “Escala de petabytes”.
12. Para el almacenamiento EBS, configure los siguientes ajustes adicionales. Es posible que algunos ajustes no aparezcan en función del tipo de volumen que elija.

Opción	Descripción
Tipo de volumen de EBS	Elija entre Propósito general (SSD) - gp3 y Propósito general (SSD) - gp2, o la generación anterior IOPS aprovisionadas (SSD) y Magnético (estándar).
Tamaño de almacenamiento de EBS por nodo	Ingrese el tamaño del volumen de EBS que desea adjuntar a cada nodo de datos. Tamaño de volumen de EBS es por cada nodo. Puede calcular el tamaño total del clúster del dominio de OpenSearch Service al multiplicar el número de nodos de datos por el tamaño del volumen de EBS. El tamaño mínimo y máximo de un volumen de EBS depende tanto del tipo de volumen de EBS especificado como del tipo de instancia al que se ha adjuntado. Para

Opción	Descripción
	obtener más información, consulte las restricciones de tamaño del volumen de EBS.
Provisioned IOPS (IOPS aprovisionadas	Si ha seleccionado un tipo de volumen IOPS aprovisionadas de SSD, ingrese el número de operaciones de E/S por segundo (IOPS) que puede soportar el volumen.

13. (Opcional) Si ha seleccionado un tipo de volumen gp3, amplíe Configuración avanzada y especifique IOPS adicionales (hasta 16 000 por cada 3 TiB de tamaño de volumen aprovisionado por nodo de datos) y rendimiento (hasta 1000 MiB/s por cada 3 TiB de tamaño de volumen aprovisionado por nodo de datos) para aprovisionar para cada nodo, más allá de lo que se incluye con el precio del almacenamiento, por un costo adicional. Para obtener más información, consulte los Precios de Amazon OpenSearch Service.
14. (Opcional) Para habilitar el almacenamiento UltraWarm, elija Habilitar nodos de datos UltraWarm. Cada tipo de instancia tiene una cantidad máxima de almacenamiento que puede poner a disposición. Multiplique esa cantidad por el número de nodos de datos templados para obtener el almacenamiento templado total a disposición.
15. (Opcional) Para habilitar el almacenamiento en frío, seleccione Habilitar almacenamiento en frío. Debe habilitar el almacenamiento UltraWarm para poder habilitar el almacenamiento en frío.
16. Si utiliza Multi-AZ con modo de espera, ya hay tres nodos maestros dedicados habilitados. Seleccione el tipo de nodos maestros que desee. Si eligió un dominio Multi-AZ sin modo de espera, seleccione Habilitar nodos maestros dedicados y elija el tipo y la cantidad de nodos maestros que desee. Los nodos maestros dedicados aumentan la estabilidad de los clústeres y son necesarios en los dominios que tienen un número de instancias mayor de 10. Recomendamos utilizar tres nodos maestros dedicados en los dominios de producción.

> **Note**
>
> Puede elegir tipos de instancias diferentes para los nodos maestros dedicados y los nodos de datos. Por ejemplo, para los nodos de datos, puede elegir instancias de uso general u optimizadas para el almacenamiento y, para los nodos maestros dedicados, instancias optimizadas para computación.

17. (Opcional) En el caso de los dominios que ejecutan OpenSearch o Elasticsearch 5.3 y posterior, la configuración de instantáneas es irrelevante. Para obtener más información sobre las instantáneas automatizadas, consulte the section called “Crear instantáneas de índice”.
18. Si desea utilizar un punto de conexión personalizado en lugar del estándar de `https://search-mydomain-1a2a3a4a5a6a7a8a9a0a9a8a7a.us-east-1.es.amazonaws.com`, seleccione Habilitar punto de enlace personalizado y proporcione un nombre y un certificado. Para obtener más información, consulte the section called “Creación de un punto de conexión personalizado”.
19. En Red, seleccione Acceso a la VPC o Acceso público. Si elige Acceso público, vaya al siguiente paso. Si elige Acceso a la VPC, asegúrese de que se cumplen los requisitos previos y, a continuación, haga lo siguiente:

Opción	Descripción
de VPC	Seleccione el ID de la nube privada virtual (VPC) que desee utilizar. La VPC y el dominio deben estar en la misma Región de AWS y debe seleccionar una VPC con tenencia establecida en Predeterminado. OpenSearch Service aún no admite VPC que utilicen la tenencia dedicada.
Subred	Seleccione una subred. Si habilitó Multi-AZ, debe elegir dos o tres subredes. OpenSearch Service situará un punto de conexión de VPC e interfaces de red elásticas en las subredes. Debe reservar suficientes direcciones IP para las interfaces de red en la subred o subredes. Para obtener más información, consulte Reserva de direcciones IP en una subred de una VPC.
Grupos de seguridad	Seleccione uno o más grupos de seguridad de VPC que permitan que la aplicación requerida llegue al dominio de OpenSearch Service en los puertos (80 o 443) y protocolos (HTTP o HTTPS) que expone el dominio. Para obtener más información, consulte the section called “Compatibilidad con VPC”.
Rol de IAM	Conserve el rol predeterminado. OpenSearch Service utiliza este rol predefinido (también conocido como rol vinculado a un servicio) para tener acceso a la VPC y colocar un punto de conexión de VPC e

Opción	Descripción
	interfaces de red en la subred de la VPC. Para obtener más información, consulte Roles vinculados a servicios para el acceso de VPC.
Tipo de dirección IP	Seleccione el tipo de dirección IP de doble pila o IPv4. La doble pila le permite compartir los recursos del dominio entre los tipos de direcciones IPv4 e IPv6, y es la opción recomendada. Si configura el tipo de dirección IP en doble pila, no podrá cambiarlo más adelante.

20. Habilite o deshabilite el control de acceso detallado:

    - Si desea utilizar IAM para la gestión de usuarios, seleccione Establecer ARN de IAM como usuario maestro y especifique el ARN para un rol de IAM.
    - Si desea utilizar la base de datos de usuarios interna, seleccione Crear usuario maestro y especifique un nombre de usuario y una contraseña.

    Cualquiera que sea la opción elegida, el usuario maestro puede acceder a todos los índices del clúster y a todas las API de OpenSearch. Para obtener información sobre qué opción elegir, consulte the section called “Conceptos clave”.

    Si deshabilita el control de acceso detallado, puede controlar el acceso a su dominio al colocarlo dentro de una VPC, al aplicar una política de acceso restrictiva o puede utilizar ambas opciones. Debe habilitar el cifrado de nodo a nodo y el cifrado en reposo para utilizar el control de acceso detallado.

    > **Note**
    >
    > Se recomienda encarecidamente habilitar un control de acceso detallado para proteger los datos del dominio. El control de acceso detallado proporciona seguridad en los niveles de clúster, índice, documento y campo.

21. (Opcional) Si desea utilizar autenticación SAML para OpenSearch Dashboards, seleccione Habilitar la autenticación SAML y configure las opciones de SAML del dominio. Para obtener instrucciones, consulte the section called “Autenticación SAML para OpenSearch Dashboards”.
22. (Opcional) Si desea utilizar la autenticación de Amazon Cognito para OpenSearch Dashboards, elija Enable Amazon Cognito authentication (Habilitación de la autenticación de Amazon Cognito). A continuación, elija el grupo de usuarios y el grupo de identidades de Amazon

Cognito que desea utilizar para la autenticación de OpenSearch Dashboards. Para obtener orientación sobre cómo crear estos recursos, consulte the section called “Descripción de la autenticación de Amazon Cognito para OpenSearch Dashboards”.

23. (Opcional) Si desea usar la autenticación del Centro de identidades de IAM (IDC) para conectar su origen de identidad existente y ofrecer a sus aplicaciones de AWS una vista común de sus usuarios, elija Habilitar acceso a la API autenticado con el Centro de identidades de IAM. Para obtener más información, consulte Descripción general de la propagación de identidades confiables en la Guía del usuario de Centro de identidades de IAM.
24. (Opcional) En la sección Característica avanzadas, deje seleccionadas Habilitar generación de consultas en lenguaje natural y características de Amazon Q Developer, si desea utilizar estas funciones.

    Seleccione Activar S3 Vectors como opción de motor para mejorar las opciones de búsqueda vectorial. Para obtener más información, consulte (Vista previa) Capacidades de búsqueda avanzada con un motor de Amazon S3 Vectors.
25. En Política de acceso, seleccione una política de acceso o configure una de las que posee. Si decide crear una política personalizada, puede configurarla usted mismo o importar una de otro dominio. Para obtener más información, consulte the section called “Identity and Access Management”.

    > **Note**
    >
    > Si habilitó el acceso de la VPC, no puede utilizar políticas basadas en IP. En su lugar, puede utilizar grupos de seguridad para controlar qué direcciones IP pueden tener acceso al dominio. Para obtener más información, consulte the section called “Acerca de las políticas de acceso en los dominios de VPC”.
26. (Opcional) Para determinar que todas las solicitudes al dominio lleguen a través de HTTPS, seleccione Require HTTPS for all traffic to the domain (Solicitar HTTPS para todo el tráfico del dominio). Para habilitar el cifrado de nodo a nodo, seleccione Cifrado de nodo a nodo. Para obtener más información, consulte the section called “Cifrado de nodo a nodo”. Para habilitar el cifrado de los datos en reposo, seleccione Habilitar cifrado de datos en reposo. Estas opciones están preseleccionadas si elige la opción de implementación de Multi-AZ con modo de espera.
27. (Opcional) Seleccione Utilizar clave propia de AWS para que OpenSearch Service cree una clave de cifrado AWS KMS en su nombre (o bien utilice la que ya haya creado). De lo contrario,

elija su propia clave de KMS. Para obtener más información, consulte the section called “Cifrado en reposo”.

28. Para las Ventanas de menor actividad, seleccione una hora de inicio para programar las actualizaciones del software del servicio y las optimizaciones de ajuste automático que requieran una implementación azul/verde. Las actualizaciones en ventanas de menor actividad ayudan a minimizar la carga de los nodos maestros dedicados de un clúster durante los períodos de alto tráfico.
29. En Ajuste automático, seleccione si desea permitir que OpenSearch Service realice sugerencias al dominio de cambios de configuración relacionados con la memoria para mejorar la velocidad y la estabilidad. Para obtener más información, consulte the section called “Ajuste automático”.

    (Opcional) Seleccione Ventana de menor actividad para programar una ventana periódica para que el ajuste automático actualice el dominio.
30. (Opcional) Seleccione Actualización automática de software para activar las actualizaciones automáticas de software.
31. (Opcional) Agregue etiquetas para describir el dominio para que pueda categorizar y filtrar esa información. Para obtener más información, consulte the section called “Etiquetado de dominios”.
32. (Opcional) Expanda y configure Advanced cluster settings (Configuración avanzada del clúster). Para obtener un resumen de estas opciones, consulte the section called “Configuración avanzada de clústeres”.
33. Seleccione Crear.

Creación de dominios de OpenSearch Service (AWS CLI)

En lugar de crear un dominio de OpenSearch Service mediante la consola, puede utilizar la AWS CLI. Para obtener información sobre la sintaxis, consulte Amazon OpenSearch Service en la Referencia de comandos de la AWS CLI.

Comandos de ejemplo

Este primer ejemplo ilustra la siguiente configuración de un dominio de OpenSearch Service:

- Crea un dominio de OpenSearch Service llamado mylogs con OpenSearch versión 1.2
- Rellena el dominio con dos instancias del tipo `r6g.large.search`.

- Utiliza un volumen de EBS de uso general (SSD) gp3 de 100 GiB para el almacenamiento de cada nodo de datos
- Permite el acceso anónimo, pero solo desde una única dirección IP: 192.0.2.0/32

```
aws opensearch create-domain \
    --domain-name mylogs \
    --engine-version OpenSearch_1.2 \
    --cluster-config  InstanceType=r6g.large.search,InstanceCount=2 \
    --ebs-options
 EBSEnabled=true,VolumeType=gp3,VolumeSize=100,Iops=3500,Throughput=125 \
    --access-policies '{"Version": "2012-10-17",        "Statement": [{"Action":
 "es:*", "Principal":"*","Effect": "Allow", "Condition": {"IpAddress":{"aws:SourceIp":
["192.0.2.0/32"]}}}]}'
```

El próximo ejemplo ilustra la siguiente configuración de un dominio de OpenSearch Service:

- Crea un dominio de OpenSearch Service llamado mylogs con Elasticsearch versión 7.10
- Rellena el dominio con seis instancias del tipo `r6g.large.search`.
- Utiliza un volumen de EBS de uso general (SSD) gp2 de 100 GiB para el almacenamiento de cada nodo de datos
- Restringe el acceso al servicio a un solo usuario, identificado por el ID de Cuenta de AWS del usuario: 555555555555
- Distribuye las instancias entre tres zonas de disponibilidad.

```
aws opensearch create-domain \
    --domain-name mylogs \
    --engine-version Elasticsearch_7.10 \
    --cluster-config
 InstanceType=r6g.large.search,InstanceCount=6,ZoneAwarenessEnabled=true,ZoneAwarenessConfig={A
 \
    --ebs-options EBSEnabled=true,VolumeType=gp2,VolumeSize=100 \
    --access-policies '{"Version": "2012-10-17",        "Statement": [ { "Effect":
 "Allow", "Principal": {"AWS": "arn:aws:iam::555555555555:root" }, "Action":"es:*",
 "Resource": "arn:aws:es:us-east-1:555555555555:domain/mylogs/*" } ] }'
```

El próximo ejemplo ilustra la siguiente configuración de un dominio de OpenSearch Service:

- Crea un dominio de OpenSearch Service llamado mylogs con OpenSearch versión 1.0

- Rellena el dominio con diez instancias del tipo `r6g.xlarge.search`.
- Rellena el dominio con tres instancias del tipo `r6g.large.search` que actuarán como nodos maestros dedicados.
- Usa un volumen de EBS de IOPS aprovisionadas de 100 GiB para el almacenamiento, configurado con una velocidad de referencia de 1000 IOPS por cada nodo de datos
- Restringe el acceso a un único usuario y a un único subrecurso, la API `_search`

```
aws opensearch create-domain \
    --domain-name mylogs \
    --engine-version OpenSearch_1.0 \
    --cluster-config
 InstanceType=r6g.xlarge.search,InstanceCount=10,DedicatedMasterEnabled=true,DedicatedMasterTyp
 \
    --ebs-options EBSEnabled=true,VolumeType=io1,VolumeSize=100,Iops=1000 \
    --access-policies '{"Version": "2012-10-17",        "Statement": [ { "Effect":
 "Allow", "Principal": { "AWS": "arn:aws:iam::555555555555:root" }, "Action": "es:*",
 "Resource": "arn:aws:es:us-east-1:555555555555:domain/mylogs/_search" } ] }'
```

> **Note**
>
> Si intenta crear un dominio de OpenSearch Service y ya existe un dominio con el mismo nombre, la CLI no informará ningún error. En su lugar, muestra los detalles sobre el dominio existente.

Creación de dominios de OpenSearch Service (SDK de AWS)

Los AWS SDK (excepto los SDK de Android e iOS) admiten todas las acciones definidas en la Referencia de la API de configuración de Amazon OpenSearch Service, incluida `CreateDomain`. Para ver el código de muestra, consulte the section called “Utilizar los SDK de AWS”. Para obtener más información acerca de cómo instalar y utilizar los SDK de AWS, consulte los kits de desarrollo de software de AWS.

Creación de dominios de OpenSearch Service (AWS CloudFormation)

OpenSearch Service está integrado a AWS CloudFormation, un servicio que ayuda a modelar y configurar sus recursos de AWS para que pueda dedicar menos tiempo a crear y administrar

recursos e infraestructuras. Puede crear una plantilla que describa el dominio de OpenSearch que desea crear y CloudFormation aprovisiona y configura el dominio por usted. Para obtener más información, incluidos ejemplos de plantillas JSON y YAML para dominios de OpenSearch, consulte la Referencia del tipo de recurso de Amazon OpenSearch Service en la Guía del usuario de AWS CloudFormation.

Configuración de políticas de acceso

Amazon OpenSearch Service ofrece varias maneras de configurar el acceso a sus dominios de OpenSearch Service. Para obtener más información, consulte the section called “Identity and Access Management” y the section called “Control de acceso detallado”.

La consola proporciona políticas de acceso preconfiguradas que puede personalizar según las necesidades específicas de su dominio. También puede importar políticas de acceso de otros dominios de OpenSearch Service. Para obtener información sobre cómo interactúan estas políticas de acceso con el acceso de las VPC, consulte the section called “Acerca de las políticas de acceso en los dominios de VPC”.

Para configurar políticas de acceso (consola)

1. Visite https://aws.amazon.com y elija Iniciar sesión en la consola.
2. En Análisis, seleccione Amazon OpenSearch Service.
3. En el panel de navegación, en Domains (Dominios), elija el dominio que desea actualizar.
4. Seleccione Acciones y Editar la configuración de seguridad.
5. Edite la política de acceso JSON o importe una opción preconfigurada.
6. Elija Guardar cambios.

Migración a OpenSearch Service con el Asistente de Migración

El Asistente de Migración para Amazon OpenSearch Service es una solución integral que simplifica el proceso de migración desde clústeres de Elasticsearch u OpenSearch autoadministrados a OpenSearch Service. Este conjunto de herramientas aborda la complejidad operativa de las migraciones, al tiempo que garantiza la integridad de los datos y valida el rendimiento después de la migración.

Descripción general

Ya sea que esté configurando una prueba de concepto en AWS, trasladando cargas de trabajo de producción o actualizando a las versiones más recientes de OpenSearch, el Asistente de Migración ofrece orientación paso a paso, mejores prácticas y herramientas para aprovechar al máximo el potencial de las migraciones de OpenSearch.

El Asistente de Migración ofrece los siguientes beneficios clave:

- Migración de metadatos: migra los metadatos del clúster, incluidos la configuración de índices, los mapeos de tipos, las plantillas de índices y los alias.
- Migración de datos: migra los datos existentes desde clústeres heredados a dominios de OpenSearch Service.
- Administración de tráfico en vivo: intercepta y redirige el tráfico en vivo desde clústeres autoadministrados a dominios de OpenSearch Service con mínima latencia.
- Replicación de tráfico: replica el tráfico de producción en los clústeres de destino para validar la precisión y el rendimiento.
- Pruebas de rendimiento: simula tráfico del mundo real capturando y reproduciendo patrones de solicitudes para refinar el rendimiento del sistema.
- Disponibilidad global: se implementa en las regiones de AWS más comunes para alcanzar cobertura global y escalabilidad.

El Asistente de Migración admite migraciones desde Elasticsearch versiones 6.x y 7.x y OpenSearch 1.x y 2.x. Para obtener más información, consulte Rutas de migración compatibles.

> **Note**
>
> Tenga en cuenta la siguiente información adicional sobre el Asistente de Migración.
>
> - La herramienta admite migraciones de múltiples saltos (por ejemplo, migración de Elasticsearch 5.x a OpenSearch Service 3.x en un solo salto).
> - Puede revertir una migración.
> - Para algunos casos de uso, la herramienta requiere poco o ningún tiempo de inactividad.
> - La herramienta proporciona relleno de datos de alto rendimiento sin afectar al clúster de origen.

Escenarios de migración

El Asistente de Migración está diseñado para manejar los siguientes escenarios de migración:

Migración de metadatos

Migra los metadatos del clúster, como la configuración de índices, alias y plantillas, desde su clúster de origen al dominio de OpenSearch Service de destino.

Migración de relleno de datos

Migra los datos existentes o históricos desde un clúster de origen a un dominio de OpenSearch Service de destino, asegurando que toda su información importante se conserve durante la transición.

Migración de tráfico en vivo

Replica el tráfico en vivo continuo desde su clúster de origen al dominio de OpenSearch Service de destino, lo que le permite mantener la disponibilidad del servicio durante la migración.

> **Important**
>
> Las estrategias de migración no son universalmente aplicables. El Asistente de Migración proporciona orientación basada en las mejores prácticas de ingeniería, pero usted debe evaluar sus requisitos específicos y realizar pruebas exhaustivas antes de migrar cargas de trabajo de producción.

Introducción del Asistente de Migración

El Asistente de Migración para Amazon OpenSearch Service está disponible como una solución de AWS con documentación completa, plantillas de implementación y código fuente. Para comenzar con el Asistente de Migración:

Para acceder a los recursos del Asistente de Migración:

1. Revise la documentación completa de la solución en Descripción general de la solución Asistente de Migración para Amazon OpenSearch Service.
2. Comprenda los costos y requisitos leyendo la guía de planificación de la implementación.

3. Despliegue la solución siguiendo las instrucciones de implementación y utilice la guía de uso para realizar su migración.

Para desarrolladores y usuarios avanzados, el código fuente del Asistente de Migración y documentación adicional están disponibles en el repositorio de GitHub de migraciones de OpenSearch.

Arquitectura de soluciones

Cuando se implementa en AWS, el Asistente de Migración utiliza varios servicios de AWS para ofrecer una solución de migración integral:

- AWS CloudFormation: proporciona plantillas de Infraestructura como Código (IaC) para implementar y configurar el Asistente de Migración.
- OpenSearch Service: el servicio de destino para sus cargas de trabajo de búsqueda y análisis migradas
- Amazon Managed Streaming para Apache Kafka: proporciona capacidades de procesamiento de flujo para almacenamiento duradero y reutilización del tráfico HTTP.
- Amazon Elastic Container Service: ejecuta la Consola de Administración de Migraciones y el Reproductor de Tráfico en contenedores seguros y escalables.
- Amazon Elastic File System: proporciona almacenamiento persistente y escalable para los datos de solicitudes y respuestas de los clústeres de origen y destino.
- Amazon Simple Storage Service: almacena instantáneas para tareas de relleno histórico y contenido de Infraestructura como Código.

Para obtener información detallada sobre la arquitectura, consulte Detalles de la arquitectura del Asistente de Migración.

Configuración avanzada de clústeres

Use opciones avanzadas para configurar lo siguiente:

Índices en los cuerpos de las solicitudes

Especifica si se permiten referencias explícitas a índices en el cuerpo de las solicitudes HTTP. Al configurar esta propiedad en `false` se evita que los usuarios eludan el control de acceso para

subrecursos. De forma predeterminada, el valor es `true`. Para obtener más información, consulte the section called “Opciones avanzadas y consideraciones de la API”.

Asignación de una caché de datos de campos

Especifica el porcentaje de espacio de montón de Java que se asignará a datos de campos. De forma predeterminada, esta configuración es el 20 % de la pila de JVM.

> **Note**
>
> Muchos clientes consultan índices diarios en rotación. Recomendamos empezar a realizar ensayos de referencia con el valor `indices.fielddata.cache.size` configurado en un 40 % de la pila de JVM para la mayoría de estos casos de uso. Si tiene índices muy grandes es posible que necesite una caché de datos de campos de gran tamaño.

Recuento máximo de cláusulas

Especifica el número máximo de cláusulas permitidas en una consulta de Lucene con operadores booleanos. El valor predeterminado es 1,024. Las consultas que superen el número máximo permitido de cláusulas provocarán un error `TooManyClauses`. Para obtener más información, consulte la documentación de Lucene.

Cambios de configuración en Amazon OpenSearch Service

Amazon OpenSearch Service utiliza un proceso de implementación azul/verde al actualizar los dominios. Una implementación azul/verde crea un entorno inactivo para actualizaciones de dominio que copia el entorno de producción y dirige usuarios al nuevo entorno una vez completadas esas actualizaciones. En una implementación azul/verde, el entorno azul es el entorno de producción actual. El entorno verde es el entorno de almacenamiento inactivo.

Los datos se migran del entorno azul al entorno verde. Cuando el nuevo entorno está listo, OpenSearch Service conmuta los entornos para promover el entorno verde a ser el nuevo entorno de producción. La transición se produce sin pérdida de datos. Esta práctica minimiza el tiempo de inactividad y mantiene el entorno original en caso de que la implementación en el nuevo entorno no se produzca correctamente.

Temas

- Cambios que suelen causar implementaciones azul/verde
- Cambios que no suelen causar implementaciones azul/verde
- Cómo determinar si un cambio provocará una implementación azul/verde
- Seguimiento de un cambio de configuración
- Etapas de un cambio de configuración
- Impacto en el rendimiento de las implementaciones azul/verde
- Cargos por los cambios de configuración
- Solución de errores de validación

Cambios que suelen causar implementaciones azul/verde

Las siguientes operaciones provocan implementaciones azul/verde:

- Cambio del tipo de instancia
- Habilitar el control de acceso detallado
- Realización de actualizaciones de software del servicio
- Activación o desactivación de nodos maestros dedicados
- Activación o desactivación de Multi-AZ sin modo de espera
- Cambio del tipo de almacenamiento, el tipo de volumen o el tamaño de volumen
- Selección de diferentes subredes de VPC
- Agregar o eliminar grupos de seguridad de la VPC
- Agregar o eliminar nodos coordinadores dedicados
- Habilitación o deshabilitación de la autenticación de Amazon Cognito para OpenSearch Dashboards
- Elección de otro grupo de usuarios o grupo de identidades de Amazon Cognito
- Modificación de la configuración avanzada
- Actualización a una nueva versión de OpenSearch (es posible que OpenSearch Dashboards no esté disponible durante toda la actualización o parte de ella)
- Habilitación del cifrado de datos en reposo o del cifrado de nodo a nodo
- Habilitación o deshabilitación del almacenamiento en frío o del almacenamiento UltraWarm

- Desactivación de Auto-Tune y restauración de los cambios asociados
- Asociar un complemento opcional a un dominio y disociar un complemento opcional de un dominio
- Aumento del recuento de nodos maestros dedicados para dominios Multi-AZ con dos nodos maestros dedicados
- Reducción del tamaño del volumen de EBS
- Cambio del tamaño del volumen, las IOPS o el rendimiento de EBS, si el último cambio que hizo está en curso o se produjo hace menos de 6 horas
- Activación de la publicación de registros de auditoría en CloudWatch.

En el caso de los dominios Multi-AZ con modo de espera, solo puede realizar una solicitud de cambio a la vez. Si ya hay un cambio en curso, se rechaza la nueva solicitud. Puede comprobar el estado del cambio actual con la API de la `DescribeDomainChangeProgress`.

Cambios que no suelen causar implementaciones azul/verde

En la mayoría de los casos, las siguientes operaciones no provocan implementaciones azul/verde:

- Modificación de la política de acceso
- Modificación del punto de conexión personalizado
- Cambio de la política de seguridad de la capa de transporte (TLS)
- Cambio de la hora de instantánea automatizada
- Habilitar o deshabilitar Require HTTPS
- Activación de Auto-Tune o desactivación sin revertir sus cambios
- Cambio del nodo de datos o el recuento de nodos UltraWarm, si el dominio tiene nodos maestros dedicados
- Cambio del tipo o recuento de instancias maestras dedicadas (excepto en el caso de los dominios Multi-AZ con dos nodos maestros dedicados), si el dominio tiene nodos maestros dedicados
- Activación o desactivación de la publicación de registros de errores o registros lentos en CloudWatch
- Desactivación de la publicación de registros de auditoría en CloudWatch
- Aumento del tamaño del volumen hasta 3 TiB por nodo de datos, cambio del tipo de volumen, las IOPS o el rendimiento
- Cómo agregar y eliminar etiquetas

> **Note**
>
> Existen algunas excepciones en función de la versión del software de servicio. Si quiere tener la seguridad de que un cambio no causará ninguna implementación azul/verde, lleve a cabo una ejecución de prueba antes de actualizar el dominio, si esta opción está disponible. Algunos cambios no ofrecen una opción de ejecución de prueba. Por lo general, le recomendamos hacer cambios en el clúster fuera de las horas de mayor tráfico.

Cómo determinar si un cambio provocará una implementación azul/verde

Puede probar algunos tipos de cambios de configuración planificados para determinar si provocarán una implementación azul/verde sin necesidad de confirmarlos. Antes de iniciar un cambio de configuración, utilice la consola o una de API para realizar una comprobación de validación con el fin de asegurarse de que el dominio sea apto para una actualización.

Console

Validación de un cambio de configuración

1. Vaya a la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/.
2. En el panel de navegación izquierdo, seleccione Dominios.
3. Seleccione el dominio para el que desee realizar un cambio de configuración. Se abrirá la página de detalles del dominio. Seleccione el menú desplegable Acciones y, a continuación, Editar configuración del clúster.
4. Realice cambios en el dominio, como cambiar el tipo de instancia o la cantidad de nodos.
5. En Análisis de prueba, elija Ejecutar. La prueba válida su cambio de configuración en busca de errores y determina si requiere una implementación azul/verde.
6. Cuando la prueba finaliza, los resultados aparecen en la parte inferior de la página, junto con un ID de prueba. El análisis indica si el cambio de configuración requiere o no una implementación azul/verde.

   Cada prueba sobrescribe la anterior. Para retener los detalles de cada prueba, guarde su ID de prueba. Las pruebas están disponibles durante 90 días o hasta que realice una actualización de configuración.
7. Para proceder con la actualización de la configuración, seleccione Guardar cambios. De lo contrario, seleccione Cancelar. Cualquiera de las opciones lo lleva de vuelta a pestaña

Configuración del clúster. En esta pestaña, se puede seleccionar Detalles de la ejecución de prueba para ver los detalles de la última ejecución de prueba. Esta página también incluye una comparación en paralelo entre la configuración anterior a la ejecución de prueba y la configuración de la ejecución de prueba.

API

Puede realizar una validación mediante una ejecución de prueba a través de la API de configuración. Para probar los cambios con la API, establezca `DryRun` en `true`, y `DryRunMode` en `Verbose`. El modo detallado ejecuta una comprobación de validación además de determinar si el cambio iniciará una implementación azul/verde. Por ejemplo, esta solicitud UpdateDomainConfig prueba el tipo de implementación que se produce como consecuencia de habilitar UltraWarm:

```
POST https://es.us-east-1.amazonaws.com/2021-01-01/opensearch/domain/my-domain/
config
{
   "ClusterConfig": {
    "WarmCount": 3,
    "WarmEnabled": true,
    "WarmType": "ultrawarm1.large.search"
   },
   "DryRun": true,
   "DryRunMode": "Verbose"
}
```

La solicitud ejecuta una comprobación de validación y devuelve el tipo de implementación que provocará el cambio, pero no lleva a cabo la actualización:

```
{
   "ClusterConfig": {
     ...
    },
   "DryRunResults": {
      "DeploymentType": "Blue/Green",
      "Message": "This change will require a blue/green deployment."
    }
}
```

Los tipos de implementación posibles son:

- `Blue/Green`: el cambio provocará una implementación azul/verde.
- `DynamicUpdate`: el cambio no provocará una implementación azul/verde.
- `Undetermined`: el dominio sigue en estado de procesamiento, por lo que no se puede determinar el tipo de implementación.
- `None`: sin cambio en la configuración.

Si la validación falla, devuelve una lista de errores de validación.

```
{
   "ClusterConfig":{
      "..."
   },
   "DryRunProgressStatus":{
      "CreationDate":"2023-01-12T01:14:33.847Z",
      "DryRunId":"db00ca39-48b2-4774-bbd3-252cf094d205",
      "DryRunStatus":"failed",
      "UpdateDate":"2023-01-12T01:14:33.847Z",
      "ValidationFailures":[
         {
            "Code":"Cluster.Index.WriteBlock",
            "Message":"Cluster has index write blocks."
         }
      ]
   }
}
```

Si el estado sigue siendo `pending`, se puede utilizar el ID de ejecución de prueba de la respuesta de UpdateDomainConfig en las siguientes llamadas a DescribeDryRunProgress para comprobar el estado de la validación.

```
GET https://es.us-east-1.amazonaws.com/2021-01-01/opensearch/domain/my-domain/
dryRun?dryRunId=my-dry-run-id
{
    "DryRunConfig": null,
    "DryRunProgressStatus": {
        "CreationDate": "2023-01-12T01:14:42.998Z",
        "DryRunId": "db00ca39-48b2-4774-bbd3-252cf094d205",
        "DryRunStatus": "succeeded",
        "UpdateDate": "2023-01-12T01:14:49.334Z",
        "ValidationFailures": null
```

```
    },
    "DryRunResults": {
        "DeploymentType": "Blue/Green",
        "Message": "This change will require a blue/green deployment."
    }
}
```

Para realizar un análisis de ejecución de prueba sin una comprobación de validación, establezca `DryRunMode` en`Basic` cuando utilice la API de configuración.

Python

El siguiente código de Python usa la API UpdateDomainConfig para realizar una comprobación de validación de ejecución en seco y, si esta se realiza correctamente, llama a la misma API sin una ejecución en prueba para iniciar la actualización. Si la comprobación no se realiza correctamente, el script imprime el error y se detiene.

```
import time
import boto3

client = boto3.client('opensearch')

response = client.UpdateDomainConfig(
    ClusterConfig={
        'WarmCount': 3,
        'WarmEnabled': True,
        'WarmCount': 123,
    },
    DomainName='test-domain',
    DryRun=True,
    DryRunMode='Verbose'
)

dry_run_id = response.DryRunProgressStatus.DryRunId

retry_count = 0

while True:

    if retry_count == 5:
        print('An error occured')
        break
```

```
   dry_run_progress_response = client.DescribeDryRunProgress('test-domain',
dry_run_id)
   dry_run_status = dry_run_progress_response.DryRunProgressStatus.DryRunStatus

   if dry_run_status == 'succeeded':
       client.UpdateDomainConfig(
           ClusterConfig={
           'WarmCount': 3,
           'WarmEnabled': True,
           'WarmCount': 123,
       })
       break

   elif dry_run_status == 'failed':
       validation_failures_list =
dry_run_progress_response.DryRunProgressStatus.ValidationFailures
       for item in validation_failures_list:
           print(f"Code: {item['Code']}, Message: {item['Message']}")
       break

   retry_count += 1
   time.sleep(30)
```

Seguimiento de un cambio de configuración

Puede solicitar un cambio de configuración a la vez o agrupar varios cambios en una sola solicitud. Utilice los campos Estado de procesamiento del dominio y Estado del cambio de configuración en la consola para realizar un seguimiento de los cambios de configuración. Espere a que el estado del dominio se convierta en `Active` antes de solicitar cambios adicionales.

Un dominio puede tener los siguientes estados de procesamiento:

- `Active`: no hay ningún cambio de configuración en curso. Puede enviar una nueva solicitud de cambio de configuración.
- `Creating`: se está creando el dominio.
- `Modifying`: se están haciendo cambios de configuración, como la adición de nuevos nodos de datos, EBS, gp3, el aprovisionamiento de IOPS o la configuración de claves de KMS.
- `Upgrading engine version`: se está actualizando la versión del motor.
- `Updating service software`: se está actualizando el software del servicio.

- `Deleting`: se está eliminando el dominio.
- `Isolated`: el dominio está suspendido.

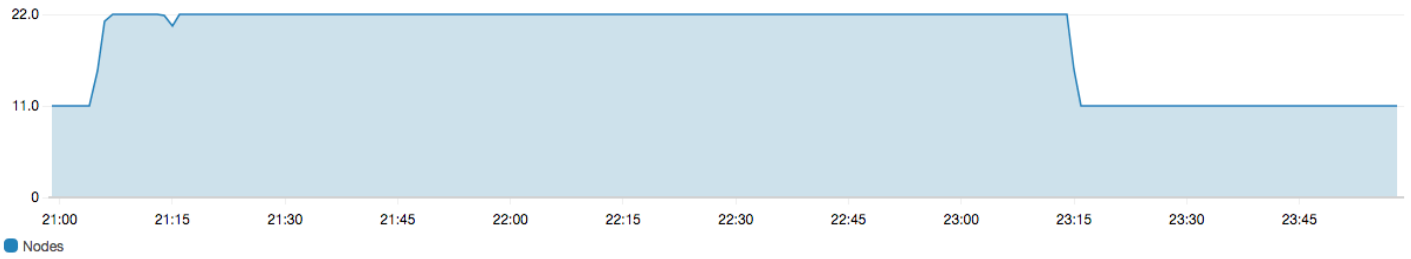
Un dominio puede tener los siguientes estados de cambio de configuración:

- `Pending`: se ha enviado una solicitud de cambio de configuración.
- `Initializing`: el servicio está iniciando un cambio de configuración.
- `Validating`: el servicio está validando los cambios solicitados y los recursos necesarios.
- `Awaiting user inputs`: el servicio espera que continúen los cambios de configuración, como un cambio de tipo de instancia. Puede editar los cambios de configuración.
- `Applying changes`: el servicio está aplicando los cambios de configuración solicitados.
- `Cancelled`: el cambio de configuración está cancelado. Seleccione Cancelar y revertir todos los cambios.
- `Completed`: los cambios de configuración solicitados se han completado correctamente.
- `Validation failed`: los cambios de configuración solicitados no se pudieron completar. No se aplicó ningún cambio de configuración.

> **Note**
>
> Los errores de validación se deben a la presencia de índices rojos en el dominio, a la falta de disponibilidad del tipo de instancia elegido o a la falta de espacio en disco. Para obtener una lista de los errores de validación, consulte the section called “Solución de errores de validación”. Durante un evento de fallo de validación, puede cancelar, reintentar o editar los cambios de configuración.

Cuando se completen los cambios de configuración, el estado del dominio volverá a ser `Active`.

P puede revisar el estado del clúster y las métricas de Amazon CloudWatch y ver que el número de nodos del clúster se incrementa temporalmente, a menudo se duplica, mientras se actualiza el dominio. En la siguiente ilustración, puede ver el número de nodos duplicarse de 11 a 22 durante un cambio de configuración y volver a 11 una vez completada la actualización.



Este duplicado provisional puede afectar al rendimiento de los nodos maestros dedicados del clúster, que repentinamente podría tener que administrar más nodos. También puede aumentar las latencias de búsqueda e indexación a medida que OpenSearch Service copia los datos del clúster anterior al nuevo. Es importante mantener una capacidad suficiente en el clúster para administrar la sobrecarga asociada a estas implementaciones azul/verde.

> **Important**
>
> El usuario no incurrirá en ningún gasto adicional durante los cambios de configuración y el mantenimiento del servicio. Solo se facturará por el número de nodos que solicite para el clúster. Para obtener información detallada, consulte the section called “Cargos por los cambios de configuración”.

Para evitar sobrecargar los nodos maestros dedicados, puede monitorear el uso con las métricas de Amazon CloudWatch. Para conocer los valores máximos recomendados, consulte the section called “Alarmas de CloudWatch recomendadas”.

Etapas de un cambio de configuración

Después de iniciar un cambio de configuración, OpenSearch Service sigue una serie de pasos para actualizar el dominio. Puede ver el progreso del cambio de configuración en Estado de cambio de configuración en la consola. Los pasos exactos por los que se realiza una actualización dependen del tipo de cambio que realices. También puede monitorear un cambio de configuración mediante la operación de la API DescribeDomainChangeProgress.

Las siguientes son las posibles etapas que puede realizar una actualización durante un cambio de configuración:

Nombre de la fase	Descripción
Validación	Validación de que el dominio es apto para una actualización y mostrar problemas de validación si es necesario.
Creación de un nuevo entorno	Completar los requisitos previos necesarios y crear los recursos necesarios para iniciar la implementación azul-verde.
Provisión de nuevos nodos	Creación de un nuevo conjunto de instancias en el nuevo entorno.
Enrutamiento del tráfico en nodos nuevos	Redireccionamiento del tráfico a los nodos de datos recién creados.

Nombre de la fase	Descripción
Enrutamiento del tráfico en nodos antiguos	Desactivación del tráfico en los nodos de datos antiguos.
Preparación de los nodos para su eliminación	Preparación para eliminar nodos. Este paso solo ocurre cuando estás reduciendo la escala de su dominio (por ejemplo, de 8 a 6 nodos).
Copia de particiones en nodos nuevos	Trasladando los fragmentos de los nodos antiguos a los nuevos.
Terminación de nodos	Finalización y eliminación de nodos antiguos después de remover los fragmentos.

Nombre de la fase	Descripción
Eliminación de recursos más antiguos	Eliminación de recursos asociados al entorno anterior (por ejemplo, equilibrador de carga).
Actualización dinámica	Se muestra cuando la actualización no requiere una implementación azul-verde y se puede aplicar dinámicamente.
Aplicación de cambios específicos relacionados con la instancia maestra	Se muestra cuando se cambia el tipo o el número de la instancia maestra dedicada.

Nombre de la fase	Descripción
Aplicación de cambios relacionados con el volumen	Se muestra cuando se modifican el tamaño, el tipo, las IOPS y el rendimiento del volumen.

Impacto en el rendimiento de las implementaciones azul/verde

Durante la implementación azul/verde, su clúster de Amazon OpenSearch Service está disponible para las solicitudes entrantes de búsqueda e indexación. Sin embargo, es posible que observe los siguientes problemas de rendimiento:

- Aumento temporal del uso en los nodos principales, ya que los clústeres tienen que administrar más nodos.
- Aumento de la latencia de búsqueda e indexación, ya que OpenSearch Service copia los datos de los nodos antiguos a los nuevos.
- Aumento de los rechazos de solicitudes entrantes, ya que la carga del clúster aumenta durante las implementaciones azul/verde.
- Para evitar problemas de latencia y rechazos de solicitudes, debe ejecutar implementaciones azul/verde cuando el clúster esté en buen estado y haya poco tráfico de red.

Cargos por los cambios de configuración

Si cambia la configuración de un dominio, OpenSearch Service crea un nuevo clúster tal y como se describe en the section called “Cambios de configuración”. Durante la migración de uno a otro, se devengarán los cargos siguientes:

- Si cambia el tipo de instancia, se cobrará por los dos clústeres durante la primera hora. Después de la primera hora, solo se cobrará por el clúster nuevo. Los volúmenes de EBS no se cobran dos veces porque forman parte del clúster, por lo que la facturación corresponde a la facturación de instancias.

Ejemplo: Supongamos que cambia la configuración y pasa de tres instancias `m3.xlarge` a cuatro instancias `m4.large`. Durante la primera hora, se cobrarán los dos clústeres (3* `m3.xlarge` + 4* `m4.large`). Después de la primera hora, solo se cobrará el clúster nuevo (4 * `m4.large`).

- Si no cambia el tipo de instancia, solo se cobrará el clúster mayor durante la primera hora. Después de la primera hora, solo se cobrará por el clúster nuevo.

  Ejemplo: Supongamos que cambia la configuración y pasa de seis instancias `m3.xlarge` a tres instancias `m3.xlarge`. Durante la primera hora, se cobrará por el clúster mayor (6 * `m3.xlarge`). Después de la primera hora, solo se cobrará el clúster nuevo (3 * `m3.xlarge`).

Solución de errores de validación

Al iniciar un cambio de configuración o una actualización de versión de OpenSearch o Elasticsearch, OpenSearch Service primero lleva a cabo una serie de comprobaciones de validación para garantizar que el dominio es apto para una actualización. Si se produce un error en alguna de estas comprobaciones, recibirá una notificación en la consola con los problemas específicos que debe corregir antes de actualizar su dominio. En la siguiente tabla, se enumeran los posibles problemas de dominio que puede presentar OpenSearch Service y los pasos para resolverlos.

Problema	Código de error	Pasos para la solución de problemas
No se encontró el grupo de seguridad	`SecurityGroupNotFound`	El grupo de seguridad asociado al dominio de OpenSearch Service no existe. Para solucionar este problema, cree un grupo de seguridad con el nombre especificado.
No se encontró la subred	`SubnetNotFound`	La subred asociada al dominio de OpenSearch Service no existe. Para solucionar este problema, cree una subred en la VPC.
No se configuró el rol vinculado al servicio	`SLRNotConfigured`	El rol vinculado al servicio de OpenSearch Service no está configurado. OpenSearch Service predefine el rol vinculado al servicio y este incluye todos los permisos que el servicio requiere para llamar a otros servicios de AWS en su nombre. Si el rol no existe, es posible que tenga que crearlo manualmente.

Problema	Código de error	Pasos para la solución de problemas
No hay suficientes direcciones IP	`InsufficientFreeIPsForSubnets`	Una o más de las subredes de VPC no tienen suficientes direcciones IP para actualizar el dominio. Para calcular el número de direcciones IP que necesita, consulte the section called “Reserva de direcciones IP en una subred de una VPC”.
No existe el grupo de usuarios de Cognito	`CognitoUserPoolNotFound`	OpenSearch Service no encuentra el grupo de usuarios de Amazon Cognito. Verifique que creó uno y que tiene el ID correcto. Para encontrar el ID, puede utilizar la consola de Amazon Cognito o el siguiente comando de la AWS CLI: `aws cognito-idp list-user-pools --max-results 60 --region us-east-1`
El grupo de identidades de Cognito no existe	`CognitoIdentityPoolNotFound`	OpenSearch Service no encuentra el grupo de identidades de Cognito. Verifique que creó uno y que tiene el ID correcto. Para encontrar el ID, puede utilizar la consola de Amazon Cognito o el siguiente comando de la AWS CLI: `aws cognito-identity list-identity-pools --max-results 60 --region us-east-1`
No se encontró el dominio de Cognito para el grupo de usuarios	`CognitoDomainNotFound`	El grupo de usuarios no tiene un nombre de dominio. Puede configurar uno con la consola de Amazon Cognito o el siguiente comando de la AWS CLI: `aws cognito-idp create-user-pool-domain --domain my-domain --user-pool-id id`

Problema	Código de error	Pasos para la solución de problemas
No se configuró el rol de Cognito	`CognitoRoleNotConfigured`	No se configuró el rol de IAM que otorga permisos a OpenSearch Service para configurar los grupos de usuarios y de identidades de Amazon Cognito, los cuales utiliza para la autenticación. Configure el rol con un conjunto de permisos y una relación de confianza adecuados. Puede utilizar la consola, que crea el rol CognitoAccessForAmazonOpenSearch predeterminado, o puede configurar manualmente un rol mediante la AWS CLI o el SDK de AWS.
No se puede describir el grupo de usuarios	`UserPoolNotDescribable`	El rol de Amazon Cognito especificado no tiene permiso para describir el grupo de usuarios asociado a su dominio. Asegúrese de que la política de permisos de rol permita la acción `cognito-identity:DescribeUserPool`. Consulte the section called “Acerca del rol CognitoAccessForAmazonOpenSearch” para ver la política de permisos completa.
No se puede describir el grupo de identidades	`IdentityPoolNotDescribable`	El rol de Amazon Cognito especificado no tiene permiso para describir el grupo de identidades asociado a su dominio. Asegúrese de que la política de permisos de rol permita la acción `cognito-identity:DescribeIdentityPool`. Consulte the section called “Acerca del rol CognitoAccessForAmazonOpenSearch” para ver la política de permisos completa.
No se pueden describir los grupos de usuarios y de identidades	`CognitoPoolsNotDescribable`	El rol de Amazon Cognito especificado no tiene permiso para describir los grupos de usuarios e identidades asociados a su dominio. Asegúrese de que la política de permisos de rol permita las acciones `cognito-identity:DescribeIdentityPool` y `cognito-identity:DescribeUserPool`. Consulte the section called “Acerca del rol CognitoAccessForAmazonOpenSearch” para ver la política de permisos completa.

Problema	Código de error	Pasos para la solución de problemas
La clave de KMS no está habilitada	`KMSKeyNotEnabled`	La clave de AWS Key Management Service (AWS KMS) utilizada para cifrar el dominio no está habilitada. Vuelva a habilitar la clave inmediatamente.
El estado del certificado personalizado no es ISSUED (EMITIDO)	`InvalidCertificate`	Si su dominio utiliza un punto de conexión personalizado, protéjalo mediante la generación de un certificado SSL en AWS Certificate Manager (ACM) o la importación de uno de los suyos. El estado del certificado debe ser Emitido. Si aparece este error, compruebe el estado del certificado en la consola de ACM. Si el estado aparece como Expired (Vencido), Failed (Error), Inactive (Inactivo) o Pending validation (Pendiente de validación), consulte la documentación sobre solución de problemas de ACM para solucionar el problema.
No hay capacidad suficiente para lanzar el tipo de instancia elegido	`InsufficientInstanceCapacity`	La capacidad del tipo de instancia solicitada no está disponible. Por ejemplo, es posible que haya solicitado cinco nodos `i3.16xlarge.search` , pero OpenSearch Service no tiene suficientes hosts de `i3.16xlarge.search` disponibles, por lo que la solicitud no se puede llevar a cabo. Compruebe los tipos de instancias admitidos en OpenSearch Service y elija un tipo de instancia distinto.
Índices rojos en clúster	`RedCluster`	Uno o más índices del clúster tienen un estado rojo, lo que lleva a un estado rojo general del clúster. Para solucionar este problema, consulte the section called “Estado rojo del clúster”.
Interruptor de circuitos de memoria, demasiadas solicitudes	`TooManyRequests`	Hay demasiadas solicitudes de búsqueda y escritura en su dominio, por lo que OpenSearch Service no puede actualizar su configuración. Puede reducir la cantidad de solicitudes, escalar las instancias verticalmente hasta 64 GiB de RAM o escalar horizontalmente mediante la adición de instancias.

Problema	Código de error	Pasos para la solución de problemas
La nueva configuración no puede almacenar datos (poco espacio en disco)	`InsufficientStorageCapacity`	El tamaño de almacenamiento configurado no puede contener todos los datos de su dominio. Para solucionar este problema, elija un volumen mayor, elimine los índices que no se utilizan o aumente la cantidad de nodos del clúster para liberar espacio en el disco de inmediato.

Particiones ancladas a nodos específicos	`ShardMovementBlocked`	Uno o más índices de su dominio están adjuntos a nodos específicos y no se pueden reasignar. Lo más probable es que esto suceda porque se configuró el filtrado de asignación de particiones, que le permite especificar qué nodos tienen permiso para alojar las particiones de un índice en particular.
		
		Para solucionar este problema, elimine los filtros de asignación de particiones de todos los índices afectados:
		
		`PUT my-index/_settings`
		`{`
		`  "settings": {`
		`  "index.routing.allocation.require._name": null`
		`  }`
		`}`

Problema	Código de error	Pasos para la solución de problemas
La nueva configuración no puede contener todas las particiones (recuento de particiones)	`TooManyShards`	El recuento de particiones de su dominio es demasiado alto, lo que impide que OpenSearch Service los mueva a la nueva configuración. Para solucionar este problema, escale su dominio horizontalmente. Para ello, agregue nodos del mismo tipo de configuración que los nodos actuales del clúster. Tenga en cuenta que el tamaño máximo de volumen de EBS depende del tipo de instancia del nodo. Para evitar este problema en el futuro, consulte the section called "Selección del número de particiones" y defina una estrategia de partición que sea adecuada para su caso de uso.
La subred asociada al dominio no admite direcciones IPv4	`ResultCodeIPv4BlockNotExists`	Para resolver este problema, cree una subred o actualice la subred existente en la VPC según el tipo de dirección IP configurada del dominio. Si su dominio usa un tipo de dirección únicamente para IPv4, use una subred solo para IPv4. Si su dominio usa el modo de doble pila, use una subred de doble pila.
La subred asociada al dominio no admite direcciones IPv6	`ResultCodeIPv6BlockNotExists`	Para resolver este problema, cree una subred o actualice la subred existente en la VPC según el tipo de dirección IP configurada del dominio. Si su dominio usa un tipo de dirección únicamente para IPv4, use una subred solo para IPv4. Si su dominio usa el modo de doble pila, use una subred de doble pila.

Actualizaciones del software de servicio en Amazon OpenSearch Service

> **Note**
>
> Para obtener una explicación de los cambios y adiciones realizados en cada actualización del software de servicio principal (sin parches), consulte las notas de la versión.

Amazon OpenSearch Service publica de forma periódica actualizaciones del software del servicio que agregan nuevas características o mejoran sus dominios. El panel Notificaciones en la consola es la forma más sencilla para ver si hay disponible una actualización o para verificar el estado de una actualización. Cada notificación incluye detalles sobre la actualización del software del servicio. Todas las actualizaciones de software utilizan implementaciones azul/verde para minimizar el tiempo de inactividad.

Las actualizaciones del software de servicio son diferentes a las actualizaciones de versión de OpenSearch. Para obtener más información acerca de cómo actualizar a una versión más reciente de OpenSearch, consulte the section called “Actualización de dominios”.

Actualizaciones opcionales frente a actualizaciones obligatorias

El Servicio de OpenSearch cuenta con dos amplias categorías de actualizaciones del software de servicio:

Actualizaciones opcionales

Las actualizaciones del software de servicio opcionales generalmente incluyen mejoras y soporte para nuevas características o funcionalidades. Las actualizaciones opcionales no se aplican a sus dominios y no hay un plazo fijo para instalarlas. La disponibilidad de la actualización se comunica por correo electrónico y una notificación en la consola. Puede optar por aplicar la actualización inmediatamente o reprogramarla para una fecha y hora más adecuadas. También puede programarla durante la ventana de menor actividad del dominio. La mayoría de las actualizaciones de software son opcionales.

Independientemente de si programa o no una actualización, si hace un cambio en el dominio que provoque una implementación azul/verde, OpenSearch Service actualizará automáticamente el software del servicio.

Puede configurar su dominio para que aplique automáticamente las actualizaciones opcionales durante la ventana de menor actividad. Cuando esta opción está activada, OpenSearch Service espera al menos 13 días desde que esté disponible una actualización opcional y, a continuación, programa la actualización después de 72 horas (tres días). Recibirá una notificación en la consola cuando la actualización esté programada y podrá elegir reprogramarla para una fecha posterior.

Para activar las actualizaciones de software automáticas, seleccione Habilitar la actualización automática de software al crear o actualizar su dominio. Para configurar el mismo ajuste mediante AWS CLI, estableza `--software-update-options` en `true` al crear o actualizar el dominio.

Actualizaciones requeridas

Las actualizaciones de software de servicio requeridas suelen incluir correcciones de seguridad críticas u otras actualizaciones obligatorias para garantizar la integridad y la funcionalidad continuas de su dominio. Algunos ejemplos de actualizaciones requeridas son las vulnerabilidades y exposiciones comunes (CVE) de Log4j y la aplicación de Servicio de metadatos de instancia Versión 2 (IMDSv2). El número de actualizaciones obligatorias en un año suele ser inferior a tres.

OpenSearch Service programa automáticamente estas actualizaciones y se lo notifica 72 horas (tres días) antes de la actualización programada mediante correo electrónico y una notificación en la consola. Puede optar por aplicar la actualización inmediatamente o reprogramarla para una fecha y hora más adecuadas dentro del plazo permitido. También puede programarla durante la próxima ventana de menor actividad del dominio. Si no realiza ninguna acción en relación con una actualización requerida y no realiza ningún cambio en el dominio que provoque una implementación azul/verde, OpenSearch Service puede iniciar la actualización en cualquier momento después de la fecha límite especificada (normalmente 14 días a partir de la disponibilidad), dentro de la ventana de menor actividad del dominio.

Independientemente de cuándo esté programada la actualización, si realiza un cambio en el dominio que provoque una implementación azul/verde, OpenSearch Service actualizará su dominio automáticamente.

Actualizaciones de revisión

Las versiones de software de servicio que terminan en “-P” y un número, como R20211203-*P4*, son versiones de revisión. Es probable que las revisiones incluyan mejoras de rendimiento, correcciones de errores menores y correcciones de seguridad o mejoras de posición. Las versiones de revisión no incluyen nuevas características ni cambios extraordinarios y, por lo general, no tienen un impacto

directo o notable en los usuarios. La notificación del software de servicio le indica si el lanzamiento de una revisión es opcional u obligatoria.

Consideraciones

Tenga en cuenta lo siguiente cuando tenga que tomar una decisión sobre la actualización del dominio:

- La actualización manual del dominio permite aprovechar las nuevas características de forma más rápida. Cuando elige Actualización, OpenSearch Service coloca la solicitud en una cola y comienza la actualización cuando tenga tiempo.
- Al iniciar una actualización del software del servicio, OpenSearch Service envía una notificación cuando se inicia la actualización y cuando esta está completa.
- Las actualizaciones de software utilizan implementaciones azul/verde para minimizar el tiempo de inactividad. Las actualizaciones pueden sobrecargar temporalmente los nodos maestros dedicados de un clúster, así que asegúrese de mantener la capacidad suficiente para manejar la sobrecarga asociada.
- Normalmente, las actualizaciones se completan en cuestión de minutos, pero también pueden tardar varias horas o incluso días si el sistema experimenta una carga pesada. Considere actualizar el dominio durante la ventana de menor actividad configurada para evitar largos periodos de actualización.

Iniciar una actualización del software de servicio

Puede solicitar una actualización del software de servicio a través de la consola de OpenSearch Service, el AWS CLI, o uno de los SDK.

Consola

Cómo solicitar una actualización del software de servicio

1. Abra la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. Seleccione el nombre de dominio para abrir su configuración.
3. Seleccione Acciones, Actualizar y seleccione una de las siguientes opciones:

    - Aplicar la actualización ahora: programa de inmediato la acción para que se lleve a cabo en la hora actual si hay capacidad disponible. Si no hay capacidad disponible, ofrecemos otras franjas horarias disponibles para elegir.

- Programarla en una ventana de menor actividad: solo está disponible si la ventana de menor actividad está activada para el dominio. Programa la actualización para que se lleve a cabo durante la ventana de menor actividad configurada en el dominio. No hay garantía de que la actualización se produzca durante la siguiente ventana inmediata. Dependiendo de la capacidad, podría producirse en los días siguientes. Para obtener más información, consulte the section called “Intervalos de menor demanda”.
- Programar para una fecha y hora específicas: programa la actualización para que se lleve a cabo en una fecha y hora específicas. Si la hora que especifica no está disponible por motivos de capacidad, puede seleccionar una franja horaria diferente.

Si programa la actualización para una fecha posterior (dentro o fuera de la ventana de menor actividad del dominio), puede reprogramarla en cualquier momento. Para obtener instrucciones, consulte the section called “Reprogramación de acciones”.

4. Seleccione Confirmar.

AWS CLI

Envíe una solicitud AWS CLI de start-service-software-update para iniciar una actualización del software del servicio. En este ejemplo, se añade la actualización a la cola de forma inmediata:

```
aws opensearch start-service-software-update \
  --domain-name my-domain \
  --schedule-at "NOW"
```

Respuesta:

```
{
    "ServiceSoftwareOptions": {
        "CurrentVersion": "R20220928-P1",
        "NewVersion": "R20220928-P2",
        "UpdateAvailable": true,
        "Cancellable": true,
        "UpdateStatus": "PENDING_UPDATE",
        "Description": "",
        "AutomatedUpdateDate": "1969-12-31T16:00:00-08:00",
        "OptionalDeployment": true
    }
}
```

> **Tip**
>
> Después de solicitar una actualización, dispone de una pequeña ventana de tiempo para cancelarla. La duración de este estado `PENDING_UPDATE` puede variar mucho y depende de su Región de AWS y el número de actualizaciones simultáneas que OpenSearch Service realice. Para cancelar una actualización, utilice la consola o el comando `cancel-service-software-update` AWS CLI.

Si la solicitud falla con una `BaseException`, eso significa que la hora que especificó no está disponible por motivos de capacidad y debe especificar una hora diferente. El servicio OpenSearch ofrece sugerencias alternativas de franjas horarias disponibles en la respuesta.

AWS SDK

Este script de Python de ejemplo utiliza los métodos describe_domain y start_service_software_update de AWS SDK para Python (Boto3) para comprobar si un dominio es apto para una actualización de software de servicio y, en caso afirmativo, iniciar la actualización. Se debe proporcionar un valor para `domain_name`.

```
import boto3
from botocore.config import Config
import time

# Build the client using the default credential configuration.
# You can use the CLI and run 'aws configure' to set access key, secret
# key, and default region.

my_config = Config(
    # Optionally lets you specify a Region other than your default.
    region_name='us-east-1'
)

domain_name = ''  # The name of the domain to check and update

client = boto3.client('opensearch', config=my_config)


def getUpdateStatus(client):
    """Determines whether the domain is eligible for an update"""
    response = client.describe_domain(
```

```
        DomainName=domain_name
    )
    sso = response['DomainStatus']['ServiceSoftwareOptions']
    if sso['UpdateStatus'] == 'ELIGIBLE':
        print('Domain [' + domain_name + '] is eligible for a service software update
 from version ' +
              sso['CurrentVersion'] + ' to version ' + sso['NewVersion'])
        updateDomain(client)
    else:
        print('Domain is not eligible for an update at this time.')


def updateDomain(client):
    """Starts a service software update for the eligible domain"""
    response = client.start_service_software_update(
        DomainName=domain_name
    )
    print('Updating domain [' + domain_name + '] to version ' +
          response['ServiceSoftwareOptions']['NewVersion'] + '...')
    waitForUpdate(client)


def waitForUpdate(client):
    """Waits for the domain to finish updating"""
    response = client.describe_domain(
        DomainName=domain_name
    )
    status = response['DomainStatus']['ServiceSoftwareOptions']['UpdateStatus']
    if status == 'PENDING_UPDATE' or status == 'IN_PROGRESS':
        time.sleep(30)
        waitForUpdate(client)
    elif status == 'COMPLETED':
        print('Domain [' + domain_name +
              '] successfully updated to the latest software version')
    else:
        print('Domain is not currently being updated.')

def main():
    getUpdateStatus(client)
```

Programación de actualizaciones del software durante las ventanas de menor actividad

Cada dominio de OpenSearch Service creado después del 16 de febrero de 2023 tiene una ventana diaria de 10 horas entre las 22:00 y las 8:00, hora local, que consideramos la ventana de menor actividad. OpenSearch Service utiliza esta ventana para programar las actualizaciones del software del servicio para el dominio. Las actualizaciones en ventanas de menor actividad ayudan a minimizar la carga de los nodos maestros dedicados de un clúster durante los períodos de mayor tráfico. OpenSearch Service no puede iniciar las actualizaciones fuera de esta ventana de 10 horas sin su consentimiento.

- En el caso de las actualizaciones opcionales, OpenSearch Service le notifica la disponibilidad de la actualización y le solicita que programe la actualización durante una ventana de menor actividad.
- En el caso de las actualizaciones requeridas, OpenSearch Service programa automáticamente la actualización durante una próxima ventana de menor actividad y se lo notifica con tres días de antelación. Puede reprogramar la actualización (dentro o fuera de la ventana de menor actividad), pero solo dentro del plazo requerido para que se complete la actualización.

Para cada dominio, puede optar por anular la hora de inicio predeterminada a las 22:00 por una hora personalizada. Para obtener instrucciones, consulte the section called “Configuración de un intervalo de menor demanda personalizado”.

Consola

Cómo programar una actualización durante una próxima ventana de menor actividad

1. Abra la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. Seleccione el nombre de dominio para abrir su configuración.
3. Seleccione Acciones, Actualizar.
4. Seleccione Programarla en la ventana de menor actividad.
5. Seleccione Confirmar.

Puede ver la acción programada en la pestaña de la ventana de menor actividad y reprogramarla en cualquier momento. Consulte the section called “Visualización de acciones programadas”.

CLI

Para programar una actualización durante una próxima ventana de menor actividad mediante el AWS CLI, envíe una solicitud de StartServicesOftwareUpdate y especifique `OFF_PEAK_WINDOW` para el parámetro `--schedule-at`:

```
aws opensearch start-service-software-update \
  --domain-name my-domain \
  --schedule-at "OFF_PEAK_WINDOW"
```

Supervisar actualizaciones de software de servicio

OpenSearch Service envía una notificación cuando hay disponible una actualización del software del servicio, obligatoria, iniciada, completada o fallada. Puede ver estas notificaciones en el panel Notificaciones de la consola de OpenSearch Service. La gravedad de la notificación es `Informational` si la actualización es opcional y `High` si es necesaria.

OpenSearch Service también envía eventos del software de servicio a Amazon EventBridge. Puede utilizar EventBridge para configurar reglas que envíen un correo electrónico o realicen una acción específica cuando se reciba un evento. Para ver un tutorial de ejemplo, consulte the section called “Tutorial: Sending SNS alerts for available updates”.

Para ver el formato de cada evento de software de servicio enviado a Amazon EventBridge, consulte the section called “Eventos de actualización del software de servicio”.

Cuando los dominios no son aptos para una actualización

El dominio no es apto para una actualización del software de servicio si está en cualquiera de los estados que se muestran a continuación:

Estado	Descripción
Dominio en procesamiento	El dominio está en medio de un cambio de configuración. Compruebe si es posible optar a la actualización una vez que se complete la operación.
Estado rojo del clúster	Uno o varios índices en el clúster aparecen en rojo. Para ver los pasos de solución de problemas, consulte the section called “Estado rojo del clúster”.

Estado	Descripción
Tasa de errores alta	El clúster de OpenSearch devuelve un gran número de errores 5xx al intentar procesar solicitudes. Este problema suele ser el resultado de demasiadas solicitudes de lectura o escritura simultáneas. Considere la posibilidad de reducir el tráfico hacia el clúster o el escalado del dominio.
Cerebro dividido	Cerebro dividido significa que el clúster de OpenSearch tiene más de un nodo maestro y se ha dividido en dos clústeres que nunca se volverán a unir por sí mismos. Puede evitar un cerebro dividido con el número recomendado de nodos maestros dedicados. Para obtener ayuda a fin de recuperar un cerebro dividido, póngase en contacto con Soporte.
Problema de integración de Amazon Cognito	Su dominio utiliza autenticación para OpenSearch Dashboards y OpenSearch Service no puede encontrar uno o más recursos de Amazon Cognito. Este problema suele producirse si no se encuentra el grupo de usuarios de Amazon Cognito. Para corregir el problema, vuelva a crear el recurso que falta y configure el dominio de OpenSearch Service para utilizarlo.
Otro problema de servicio	Los problemas con OpenSearch Service en sí pueden hacer que el dominio no pueda optar a una actualización. Si en su dominio no se aplican ninguna de las condiciones anteriores y el problema persiste durante más de un día, póngase en contacto con Soporte.

Definición de intervalos de menor demanda para Amazon OpenSearch Service

Cuando crea un dominio de Amazon OpenSearch Service, se define un intervalo diario de 10 horas que se considera como horas de menor demanda. OpenSearch Service utiliza este intervalo para programar las actualizaciones del software del servicio y las optimizaciones de ajuste automático que requieren una implementación azul/verde durante tiempos de tráfico comparativamente más bajos, siempre que sea posible. Azul/verde se refiere al proceso de crear un nuevo entorno para actualizaciones de dominio y redirigir a los usuarios al nuevo entorno una vez completadas dichas actualizaciones.

Si bien las implementaciones azul/verde no son disruptivas, para minimizar cualquier posible impacto en el rendimiento mientras se consumen los recursos de una implementación azul/verde, le recomendamos que programe estas implementaciones durante el intervalo de menor demanda configurada en el dominio. Las actualizaciones, como las sustituciones de nodos o las que deben implementarse en el dominio de forma inmediata, no utilizan el intervalo de menor demanda.

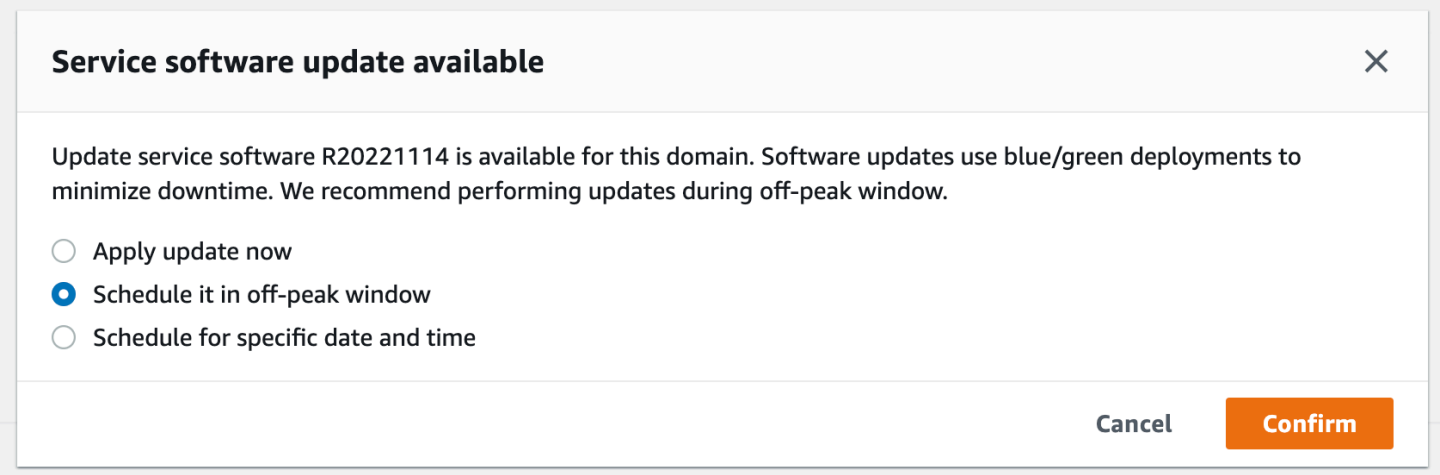
Puede modificar la hora de inicio del intervalo de menor demanda, pero no puede modificar la longitud del intervalo.

> **Note**
>
> Los intervalos de menor demanda se introdujeron el 16 de febrero de 2023. Todos los dominios creados antes de esta fecha tienen el intervalo de menor demanda desactivado de forma predeterminada. Debe habilitar y configurar manualmente el intervalo de menor demanda para estos dominios. Todos los dominios creados después de esta fecha tendrán habilitado el intervalo de menor demanda de forma predeterminada. No puede deshabilitar el intervalo de menor demanda de un dominio una vez que esté habilitado.

Actualizaciones de software de servicio de menor demanda

OpenSearch Service tiene dos amplias categorías de actualizaciones de software de servicio: opcionales y obligatorias. Ambos tipos requieren implementaciones azul/verde. Las actualizaciones opcionales no se aplican en los dominios, mientras que las actualizaciones obligatorias se instalan automáticamente si no realiza ninguna acción antes de la fecha límite especificada (normalmente dos semanas después de la disponibilidad). Para más información, consulte the section called “Actualizaciones opcionales frente a actualizaciones obligatorias”.

Cuando inicia una actualización opcional, tiene la opción de aplicarla inmediatamente, programarla para un intervalo de menor demanda posterior o especificar una fecha y hora personalizadas para aplicarla.



Para las actualizaciones obligatorias, OpenSearch Service programa automáticamente una fecha y una hora durante las horas de menor demanda para realizar la actualización. Recibe una notificación tres días antes de la actualización programada, y puede optar por reprogramarla para una fecha y hora posteriores dentro del intervalo de implementación obligatorio. Para ver las instrucciones, consulte the section called “Reprogramación de acciones”.

Optimizaciones de ajuste automático durante horas de menor demanda

Anteriormente, el ajuste automático utilizaba intervalos de mantenimiento para programar los cambios que requerían una implementación azul/verde. Los dominios que ya tenían habilitados ajuste automático y los intervalos de mantenimiento antes de la introducción de los intervalos de menor demanda seguirán utilizando los intervalos de mantenimiento para estas actualizaciones, a menos que los migre para utilizar los intervalos de menor demanda.

Le recomendamos que migre sus dominios para utilizar el intervalo de menor demanda, ya que se utiliza para programar otras actividades en el dominio, como las actualizaciones del software de servicio. Para ver las instrucciones, consulte the section called “Migración desde los intervalos de mantenimiento de ajuste automático”. No puede volver a usar los intervalos de mantenimiento después de migrar su dominio a los intervalos de menor demanda.

Todos los dominios creados después del 16 de febrero de 2023 utilizarán el intervalo de menor demanda, en lugar de los intervalos de mantenimiento tradicionales, para programar las implementaciones azul/verde. No puede deshabilitar el intervalo de menor demanda de un dominio. Para ver una lista de las optimizaciones de ajuste automático que requieren implementaciones azul/verde, consulte the section called “Tipos de cambios”.

Habilitación del intervalo de menor demanda

Todos los dominios creados antes del 16 de febrero de 2023 (cuando se introdujeron los intervalos de menor demanda) tienen esa característica desactivada de forma predeterminada. Debe habilitarlo manualmente para estos dominios. No puede deshabilitar el intervalo de menor demanda después de activarlo.

Consola

Cómo habilitar el intervalo de menor demanda en un dominio

1. Abra la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. Seleccione el nombre del dominio para abrir la configuración.
3. Acceda a la pestaña de intervalo de menor demanda y seleccione Editar.
4. Especifique una hora de inicio personalizada de acuerdo con el tiempo universal coordinado (UTC). Por ejemplo, para configurar una hora de inicio a las 23:30 en la región Oeste de EE. UU. (Oregón), especifique 07:30.
5. Seleccione Guardar cambios.

CLI

Para modificar el intervalo de menor demanda mediante la AWS CLI, envíe una solicitud UpdateDomainConfig:

```
aws opensearch update-domain-config \
  --domain-name my-domain \
  --off-peak-window-options 'Enabled=true,
 OffPeakWindow={WindowStartTime={Hours=02,Minutes=00}}'
```

Si no especifica una hora de inicio personalizada para el intervalo, el valor predeterminado es 00:00 UTC.

Configuración de un intervalo de menor demanda personalizado

Debe especificar un intervalo de menor demanda personalizado para su dominio usando el tiempo universal coordinado (UTC). Por ejemplo, si desea que el intervalo de menor demanda comience a las 23:00 para un dominio en la región Este de EE. UU. (Norte de Virginia), debe especificar a las 04:00 UTC.

Consola

Cómo modificar el intervalo de horas de menor demanda de un dominio

1. Abra la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. Seleccione el nombre del dominio para abrir la configuración.
3. Acceda a la pestaña del intervalo de menor demanda. Puede ver el intervalo de menor demanda configurado y una lista de las próximas acciones programadas para el dominio.
4. Seleccione Editar y especifique una nueva hora de inicio en UTC. Por ejemplo, para configurar la hora de inicio a las 21:00 en la región Este de EE. UU. (Norte de Virginia), especifique 02:00 UCT.
5. Seleccione Guardar cambios.

CLI

Para configurar un intervalo de menor demanda personalizado mediante la AWS CLI, envíe una solicitud UpdateDomainConfig y especifique la hora y los minutos en formato horario de 24 horas.

Por ejemplo, la siguiente solicitud cambia la hora de inicio del intervalo a las 2:00 UTC:

```
aws opensearch update-domain-config \
  --domain-name my-domain \
  --off-peak-window-options 'OffPeakWindow={WindowStartTime={Hours=02,Minutes=00}}'
```

Si no especifica la hora de inicio del intervalo, el valor predeterminado es las 22:00, hora local, para la Región de AWS en la que se crea el dominio.

Visualización de acciones programadas

Puede ver todas las acciones que están actualmente programadas, en curso o pendientes para cada uno de sus dominios. Las acciones pueden tener una gravedad de `HIGH`, `MEDIUM` y `LOW`.

Las acciones pueden tener los siguientes estados:

- `Pending update`: la acción está en la cola para ser procesada.
- `In progress`: la acción está actualmente en curso.
- `Failed`: la acción no pudo completarse.

- `Completed`: la acción ha finalizado correctamente.
- `Not eligible`: solo para actualizaciones de software de servicio. La actualización no puede continuar porque el clúster está en mal estado.
- `Eligible`: solo para actualizaciones de software de servicio. El dominio es elegible para una actualización.

Consola

La consola de OpenSearch Service muestra todas las acciones programadas en la configuración del dominio, junto con la gravedad y el estado actual de cada acción.

Cómo visualizar las acciones programadas de un dominio

1. Abra la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. Seleccione el nombre del dominio para abrir la configuración.
3. Acceda a la pestaña del intervalo de menor demanda.
4. En Acciones programadas, consulte todas las acciones que están actualmente programadas, en curso o pendientes en el dominio.

CLI

Para ver las acciones programadas mediante la AWS CLI, envíe una solicitud ListScheduledActions:

```
aws opensearch list-scheduled-actions \
  --domain-name my-domain
```

Respuesta:

```
{
    "ScheduledActions": [
        {
            "Cancellable": true,
            "Description": "The Deployment type is : BLUE_GREEN.",
            "ID": "R20220721-P13",
            "Mandatory": false,
            "Severity": "HIGH",
            "ScheduledBy": "CUSTOMER",
            "ScheduledTime": 1.673871601E9,
```

```
            "Status": "PENDING_UPDATE",
            "Type": "SERVICE_SOFTWARE_UPDATE",
        },
        {
            "Cancellable": true,
            "Description": "Amazon Opensearch will adjust the young generation JVM
 arguments on your domain to improve performance",
            "ID": "Auto-Tune",
            "Mandatory": true,
            "Severity": "MEDIUM",
            "ScheduledBy": "SYSTEM",
            "ScheduledTime": 1.673871601E9,
            "Status": "PENDING_UPDATE",
            "Type": "JVM_HEAP_SIZE_TUNING",
        }
    ]
}
```

Reprogramación de acciones

OpenSearch Service le notifica las actualizaciones programadas del software de servicio y las optimizaciones de ajuste automático. Puede optar por aplicar el cambio de inmediato o reprogramarlo para una fecha y hora posteriores.

> **Note**
>
> OpenSearch Service puede programar la acción una hora después de la hora que seleccione. Por ejemplo, si decide aplicar una actualización a las 17:00 horas, puede aplicarla entre las 17:00 y las 18:00.

Consola

Cómo reprogramar una acción

1. Abra la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. Seleccione el nombre del dominio para abrir la configuración.
3. Acceda a la pestaña del intervalo de menor demanda.
4. En Acciones programadas, seleccione la acción y elija Reprogramar.
5. Seleccione una de las siguientes opciones:

- Aplicar la actualización ahora: programa de inmediato la acción para que se lleve a cabo en la hora actual si hay capacidad disponible. Si no hay capacidad disponible, ofrecemos otras franjas horarias disponibles para elegir.
- Programarla en un intervalo de menor demanda: marca la acción para que se retome durante un próximo intervalo de menor demanda. No hay garantía de que el cambio se implemente durante el intervalo inmediatamente siguiente. Dependiendo de la capacidad, podría producirse en los días siguientes.
- Reprogramar esta actualización: permite especificar una fecha y hora personalizadas para aplicar el cambio. Si la hora que especifica no está disponible por motivos de capacidad, puede seleccionar una franja horaria diferente.
- Cancelar actualización programada: cancela la actualización. Esta opción solo está disponible para las actualizaciones de software de servicio opcionales. No está disponible para las acciones de ajuste automático ni para las actualizaciones de software obligatorias.

6. Seleccione Guardar cambios.

CLI

Para reprogramar una acción mediante la AWS CLI, envíe una solicitud UpdateScheduledAction. Para recuperar el ID de la acción, envíe una solicitud `ListScheduledActions`.

La siguiente solicitud reprograma una actualización del software de servicio para una fecha y hora específicas:

```
aws opensearch update-scheduled-action \
  --domain-name my-domain \
  --action-id R20220721-P13 \
  --action-type "SERVICE_SOFTWARE_UPDATE" \
  --desired-start-time 1677348395000 \
  --schedule-at TIMESTAMP
```

Respuesta:

```
{
   "ScheduledAction": {
      "Cancellable": true,
      "Description": "Cluster status is updated.",
      "Id": "R20220721-P13",
```

```
      "Mandatory": false,
      "ScheduledBy": "CUSTOMER",
      "ScheduledTime": 1677348395000,
      "Severity": "HIGH",
      "Status": "PENDING_UPDATE",
      "Type": "SERVICE_SOFTWARE_UPDATE"
   }
}
```

Si la solicitud falla con una `SlotNotAvailableException`, eso significa que la hora que especificó no está disponible por motivos de capacidad y debe especificar una hora diferente. El servicio OpenSearch ofrece sugerencias alternativas de franjas horarias disponibles en la respuesta.

Migración desde los intervalos de mantenimiento de ajuste automático

Si un dominio se creó antes del 16 de febrero de 2023, podría utilizar los intervalos de mantenimiento para programar las optimizaciones de ajuste automático que requieran una implementación azul/verde. Puede migrar los dominios de ajuste automático existentes para utilizar en su lugar el intervalo de menor demanda.

> **Note**
>
> No puede volver a usar los intervalos de mantenimiento después de migrar su dominio para usar los intervalos de menor demanda.

Consola

Cómo migrar un dominio para utilizar el intervalo de menor demanda

1. En la consola de Amazon OpenSearch Service, seleccione el nombre del dominio para abrir su configuración.
2. Vaya a la pestaña Ajuste automático y seleccione Editar.
3. Seleccione Migrar al intervalo de menor demanda.
4. En Hora de inicio (UTC), indique una hora de inicio diaria para el intervalo de menor demanda en el tiempo universal coordinado (UTC).
5. Seleccione Guardar cambios.

CLI

Para migrar desde un intervalo de mantenimiento de ajuste automático a un intervalo de menor demanda mediante la AWS CLI, envíe una solicitud de UpdateDomainConfig:

```
aws opensearch update-domain-config \
  --domain-name my-domain \
  --auto-tune-options
 DesiredState=ENABLED,UseOffPeakWindow=true,MaintenanceSchedules=[]
```

El intervalo de menor demanda debe estar activado para poder migrar un dominio del intervalo de mantenimiento de ajuste automático al intervalo de menor demanda. Puede activar el intervalo de menor demanda en una solicitud independiente o en la misma solicitud. Para obtener instrucciones, consulte the section called “Habilitación del intervalo de menor demanda”.

Notificaciones en Amazon OpenSearch Service

Las notificaciones de Amazon OpenSearch Service contienen información importante sobre el rendimiento y el estado de sus dominios. OpenSearch Service le notifica sobre las actualizaciones del software del servicio, las mejoras de ajuste automático, los eventos de estado del clúster y los errores de dominio. Las notificaciones están disponibles en todas las versiones de OpenSearch y Elasticsearch OSS.

Puede ver las notificaciones en el panel Notifications (Notificaciones) de la consola de OpenSearch Service. Todas las notificaciones de OpenSearch Service también aparecen en Amazon EventBridge. Para obtener una lista completa de las notificaciones y los eventos de muestra, consulte the section called “Supervisión de eventos”.

Introducción a las notificaciones

Las notificaciones se habilitan automáticamente al crear un dominio. Vaya al panel de Notifications (Notificaciones) de la consola de OpenSearch Service para monitorear y confirmar las notificaciones. Cada notificación incluye información, como la hora en que se publicó, el dominio con el que se relaciona, un nivel de severidad y estado y una breve explicación. Puede ver notificaciones históricas de hasta 90 días en la consola.

Después de acceder al panel Notifications (Notificaciones) o de confirmar la recepción de una notificación, es posible que reciba un mensaje de error en el que se indique que no tiene permisos

para realizar `es:ListNotificationsV2` o `es:UpdateNotificationStatus`. Para resolver este problema, otorgue a su usuario o rol los siguientes permisos en IAM:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": [
      "es:DescribeDomain",
      "es:ListDomainNames"
    ],
    "Resource": "arn:aws:es:*:111122223333:domain/*"
  }]
}
```

La consola de IAM genera el error “IAM does not recognize one or more actions” (“IAM no reconoce una o varias acciones”) que se puede ignorar sin riesgo. También puede restringir la acción `es:UpdateNotificationStatus` en determinados dominios. Para obtener más información, consulte the section called “Referencia de los elementos de las políticas”.

Severidad de las notificaciones

Las notificaciones en OpenSearch Service pueden ser informativas, que se refieren a cualquier acción que ya se haya realizado o a las operaciones del dominio, o procesables, que requieren que se realicen acciones específicas, como aplicar un parche de seguridad obligatorio. Cada notificación tiene una severidad asociada, que puede ser `Informational`, `Low`, `Medium`, `High` o `Critical`. En la siguiente tabla se resumen las severidades:

Gravedad	Descripción	Ejemplos
`Informational`	Información relacionada con la operación del dominio.	• Actualización del software de servicio disponible • Ajuste automático iniciado
`Low`	Una acción recomendada, pero que no tiene	• Ajuste automático cancelado

Gravedad	Descripción	Ejemplos
	ningún impacto negativo en la disponibilidad o el rendimiento del dominio si no se la realiza.	• Advertencia de alto recuento de fragmentos
`Medium`	Puede haber una consecuencia si no se realiza la acción recomendada, pero cuenta con una ventana de tiempo extendida para la acción que se va a realizar.	• Error al actualizar el software del servicio • Límite de recuento de fragmentos superado
`High`	Es necesario realizar acciones urgentes para evitar efectos adversos.	• Se requiere actualización del software de servicio • Clave KMS inaccesible
`Critical`	Se requiere una acción inmediata para evitar efectos adversos, o para recuperarse de ellos.	Ninguno disponible

Ejemplo de evento de EventBridge

En el ejemplo siguiente se muestra un evento de notificación de OpenSearch Service enviado a Amazon EventBridge. La notificación tiene una severidad `Informational` porque la actualización es opcional:

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Software Update Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
```

```
  "detail": {
    "event": "Service Software Update",
    "status": "Available",
    "severity": "Informational",
    "description": "Service software update [R20200330-p1] available."
  }
}
```

Configuración de un dominio multi-AZ en Amazon OpenSearch Service

Para evitar la pérdida de datos y minimizar el tiempo de inactividad del clúster de Amazon OpenSearch Service en caso de interrupción del servicio, puede distribuir los nodos en dos o tres zonas de disponibilidad de la misma región, una configuración conocida como Multi-AZ. Las zonas de disponibilidad son ubicaciones aisladas dentro de cada región de AWS.

Para los dominios que ejecutan cargas de trabajo de producción, recomendamos la opción de implementación de Multi-AZ con modo de espera, que crea la siguiente configuración:

- El dominio implementado en tres zonas.
- Tipo de instancia de la generación actual para los nodos maestros dedicados y los nodos de datos.
- Tres nodos maestros dedicados y tres (o un múltiplo de tres) nodos de datos.
- Al menos dos réplicas para cada índice de su dominio o un múltiplo de tres copias de datos (incluidos los nodos principales y las réplicas).

El resto de esta sección contiene explicaciones y contextos sobre estas configuraciones.

Multi-AZ con modo de espera

Multi-AZ con modo de espera es una opción de implementación para los dominios de Amazon OpenSearch Service que ofrece una disponibilidad del 99,99 %, un rendimiento uniforme para las cargas de trabajo de producción y una configuración y administración de dominios simplificadas. Cuando utilice Multi-AZ con modo de espera, los dominios son resistentes a los fallos de infraestructura, sin que ello afecte al rendimiento ni a la disponibilidad. Esta opción de implementación cumple con este estándar al exigir una serie de prácticas recomendadas, como un recuento específico de nodos de datos, un recuento de nodos maestros, un tipo de instancia, un

recuento de réplicas, la configuración de las actualizaciones de software y la activación del ajuste automático.

> **Note**
>
> Multi-AZ con modo de espera está disponible para OpenSearch versión 1.3 y posteriores y requiere regiones con al menos tres Zonas de Disponibilidad.

Al utilizar Multi-AZ con modo de espera, OpenSearch Service crea un dominio en tres zonas de disponibilidad, cada una de las cuales contiene una copia completa de los datos y los datos se distribuyen equitativamente en cada una de las zonas. Su dominio reserva los nodos de una de estas zonas como en espera, lo que significa que no atienden solicitudes de búsqueda. Cuando OpenSearch Service detecta un fallo en la infraestructura subyacente, activa automáticamente los nodos en espera en menos de un minuto. El dominio sigue atendiendo las solicitudes de indexación y búsqueda, y cualquier impacto se limita al tiempo que se tarda en realizar la conmutación por error. No hay redistribución de los datos o los recursos, por lo que el rendimiento del clúster no se ve afectado y no hay riesgo de que se degrade la disponibilidad. Multi-AZ con modo de espera está disponible sin costo adicional.

Tiene dos opciones para crear un dominio con el modo de espera en Consola de administración de AWS. En primer lugar, puede crear un dominio con el método de Creación fácil y OpenSearch Service utilizará automáticamente una configuración predeterminada, que incluye lo siguiente:

- Tres zonas de disponibilidad, una de las cuales actúa como zona de espera
- Tres nodos maestros y de datos dedicados
- El ajuste automático está habilitado en el dominio
- Almacenamiento GP3 para los nodos de datos

También puede elegir el método de Creación estándar y seleccionar Dominio con modo de espera como opción de implementación. Esto le permite personalizar su dominio sin dejar de utilizar las características clave del modo de espera, como tres zonas y tres nodos maestros. Recomendamos elegir un recuento de nodos de datos que sea múltiplo de tres (el número de zonas de disponibilidad).

Una vez que haya creado su dominio, puede ir a las páginas de detalles del dominio y, en la pestaña Configuración del clúster, confirmar que en las zonas de disponibilidad aparezca 3-AZ con modo de espera.

Si tiene problemas para migrar un dominio existente a Multi-AZ con modo de espera, consulte Error migrating to Multi-AZ with Standby en la guía de solución de problemas.

Limitaciones

Al configurar un dominio con Multi-AZ con modo de espera, tenga en cuenta las siguientes limitaciones:

- El número total de particiones de un nodo no puede superar los 1000, el número total de particiones de un clúster no puede superar los 75 000 y el tamaño de una sola partición no puede superar los 65 GB.
- Multi-AZ con modo de espera solo funciona con los tipos de instancia `m5`, `c5`, `r5`, `r6g`, `r7g`, `c6g`, `m6g`, `r6gd` y `i3`. Para obtener más información sobre los tipos de instancias admitidos, consulte Tipos de instancias admitidos.
- Solo puede utilizar un SSD de IOPS aprovisionado, un SSD de uso general (GP3) o un almacenamiento respaldado por instancias con modo de espera.
- Si habilita UltraWarm en un dominio Multi-AZ con modo de espera, la cantidad de nodos templados debe ser un múltiplo de la cantidad de zonas de disponibilidad que se utilizan.

Multi-AZ sin modo de espera

El servicio OpenSearch sigue siendo compatible con Multi-AZ sin modo de espera, lo que ofrece una disponibilidad del 99,9 %. Los nodos se distribuyen en las zonas de disponibilidad y la disponibilidad depende del número de zonas de disponibilidad y de las copias de los datos. Mientras que en el modo de espera hay que configurar el dominio según las prácticas recomendadas, sin el modo de espera puede elegir su propio número de zonas de disponibilidad, nodos y réplicas. No recomendamos esta opción a menos que tenga flujos de trabajo existentes que se verían interrumpidos al crear dominios con el modo de espera.

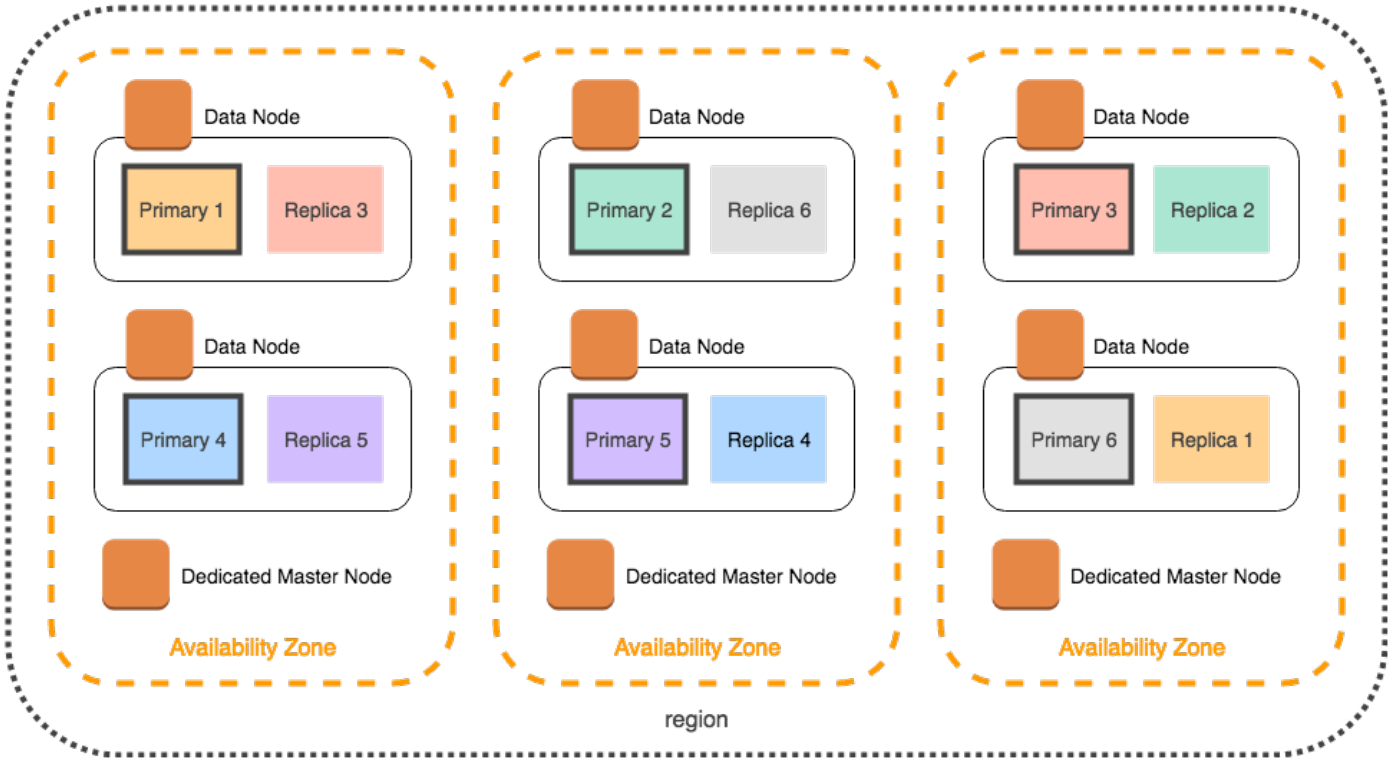
Si elige esta opción, le recomendamos que seleccione tres zonas de disponibilidad para seguir siendo resistente a los errores de nodos, discos y zonas de disponibilidad única. Cuando se produce un error, el clúster redistribuye los datos entre los recursos restantes para mantener la disponibilidad y la redundancia. Este movimiento de datos aumenta el uso de recursos en el clúster y puede afectar al rendimiento. Si el clúster no tiene el tamaño adecuado, puede experimentar una disminución de

la disponibilidad, lo que en gran medida va en contra del propósito de las zonas de disponibilidad múltiples.

La única forma de configurar un dominio sin modo de espera en el Consola de administración de AWS es elegir el método de creación Creación estándar y seleccionar Dominio sin modo de espera como opción de implementación.

Distribución de particiones

Si habilita Multi-AZ sin modo de espera, debe haber al menos una réplica por cada índice del clúster. Sin réplicas, OpenSearch Service no puede distribuir copias de sus datos a otras zonas de disponibilidad. Afortunadamente, la configuración predeterminada para cualquier índice es un recuento de réplicas de 1. Como se muestra en el siguiente diagrama, OpenSearch Service hace lo mejor posible para distribuir las particiones principales y sus particiones de réplica correspondientes entre diferentes zonas.

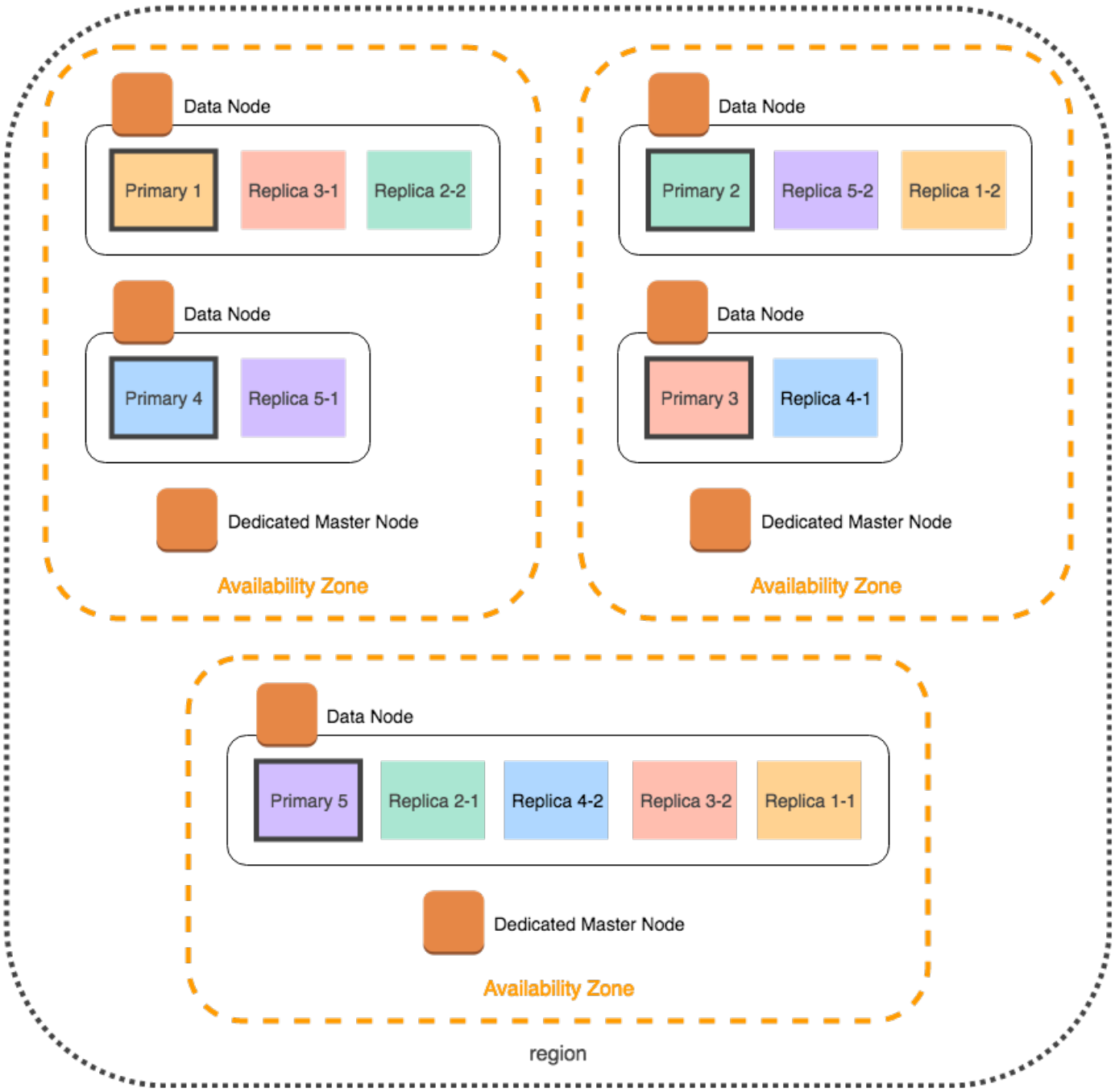


Además de distribuir las particiones por zona de disponibilidad, OpenSearch Service las distribuye también por nodo. Sin embargo, ciertas configuraciones de dominio pueden generar un número desequilibrado de particiones. Imagine el siguiente dominio:

- 5 nodos de datos

- 5 particiones principales
- 2 réplicas
- 3 zonas de disponibilidad

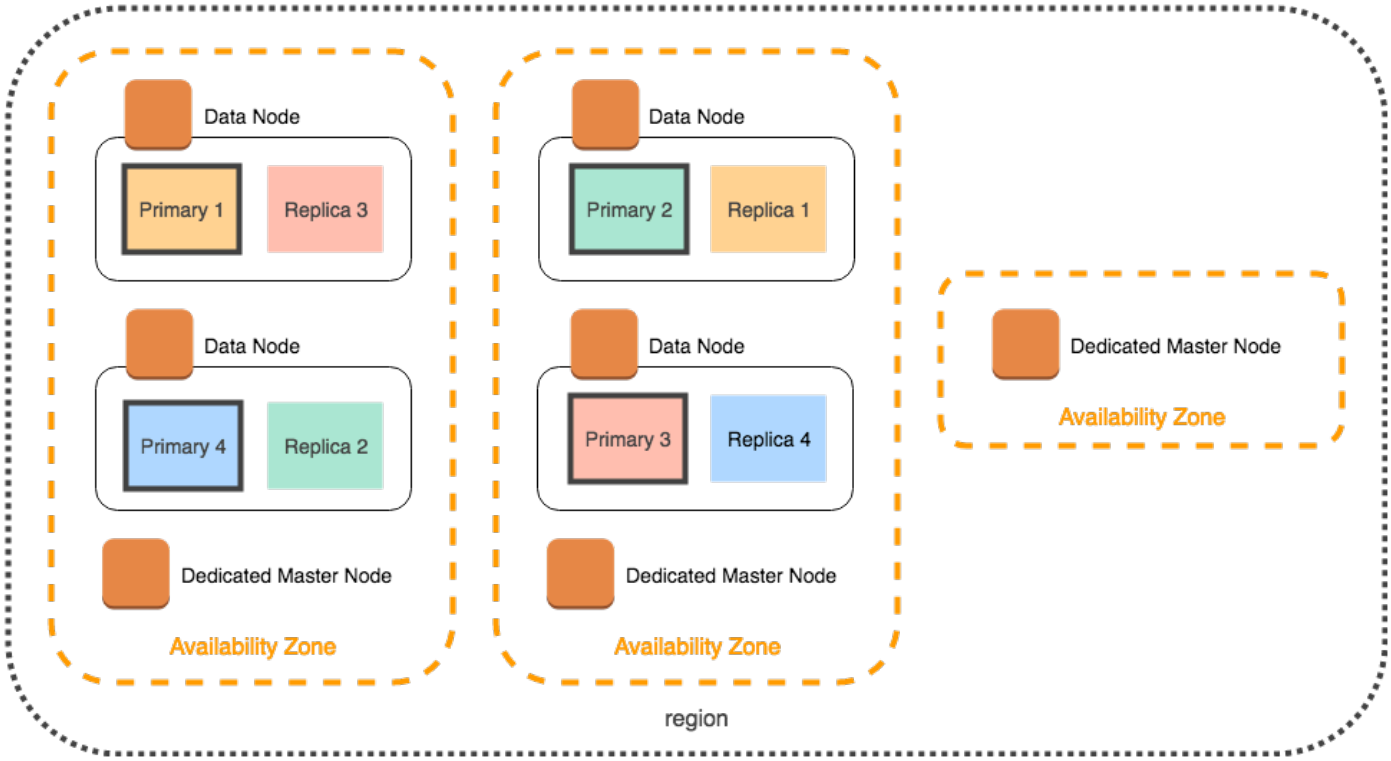
En esta situación, OpenSearch Service tiene que sobrecargar uno de los nodos para distribuir la carga de las particiones principales y replicadas entre las zonas, tal y como se muestra en el siguiente diagrama.



Para evitar este tipo de situaciones, que pueden ejercer demasiada presión sobre determinados nodos y perjudicar el rendimiento, le recomendamos que elija multi-AZ con modo de espera o que elija un número de instancias que sea múltiplo de tres cuando planee tener dos o más réplicas por cada índice.

Distribución de nodos maestros dedicados

Aunque seleccione dos zonas de disponibilidad al configurar el dominio, OpenSearch Service distribuirá automáticamente los nodos principales dedicados en tres zonas de disponibilidad. Esta distribución ayuda a evitar el tiempo de inactividad del clúster si una zona experimenta una interrupción del servicio. Si está utilizando los tres nodos principales dedicados recomendados y una zona de disponibilidad deja de funcionar, el clúster seguirá teniendo un quorum (2) de nodos principales dedicados y podrá seleccionar un nuevo nodo principal. En el siguiente diagrama, se muestra esta configuración.



Si selecciona un tipo de instancia de una generación anterior que no esté disponible en tres zonas de disponibilidad, pueden producirse las siguientes situaciones:

- Si seleccionó tres zonas de disponibilidad para el dominio, OpenSearch Service generará un error. Seleccione un tipo de instancia distinto y vuelva a intentarlo.
- Si seleccionó dos zonas de disponibilidad para el dominio, OpenSearch Service distribuirá los nodos maestros dedicados en dos zonas.

Interrupción de las zonas de disponibilidad

No es frecuente que el servicio se interrumpa en las zonas de disponibilidad, pero puede ocurrir. En la siguiente tabla, se muestran diferentes configuraciones de Multi-AZ y los comportamientos durante una interrupción. La última fila de la tabla se aplica a Multi-AZ con modo de espera, mientras que todas las demás filas tienen configuraciones que solo se aplican a Multi-AZ sin modo de espera.

Número de zonas de disponibilidad de una región	Número de zonas de disponibilidad seleccionado	Número de nodos maestros dedicados	Comportamiento si se interrumpe una zona de disponibilidad
2 o más	2	0	Tiempo de inactividad. El clúster pierde la mitad de los nodos de datos y debe sustituir al menos uno en la zona de disponibilidad que queda antes de poder elegir un nodo principal.
2	2	3	50/50 de posibilidades de tiempo de inactividad. OpenSearch Service distribuye dos nodos maestros dedicados en una zona de disponibilidad y uno en la otra: • Si la zona de disponibilidad con un nodo principal dedicado se interrumpe, los dos nodos principales dedicados de la otra zona de disponibilidad pueden elegir un nodo principal. • Si la zona de disponibilidad con dos nodos maestros dedicados experimenta una interrupción, el clúster no estará disponible hasta que la zona de disponibilidad restante se recupere.
3 o más	2	3	No hay tiempo de inactividad. OpenSearch Service distribuye automáticamente los nodos maestros dedicados entre tres zonas de disponibilidad, por lo que los dos nodos

Número de zonas de disponibilidad de una región	Número de zonas de disponibilidad seleccionado	Número de nodos maestros dedicados	Comportamiento si se interrumpe una zona de disponibilidad
			maestros dedicados pueden elegir un nodo maestro.
3 o más	3	0	No hay tiempo de inactividad. Aproximad amente dos tercios de los nodos de datos seguirán estando disponibles para elegir un nodo principal.
3 o más	3	3	No hay tiempo de inactividad. Los otros dos nodos principales dedicados pueden elegir un nodo principal.

En todas las configuraciones, independientemente de la causa, los errores en los nodos pueden hacer que los nodos de datos restantes del clúster experimenten un periodo de aumento de la carga mientras OpenSearch Service configura automáticamente nuevos nodos para sustituir a los que faltan.

Por ejemplo, en caso de que se interrumpa una zona de disponibilidad de una configuración de tres zonas, solo dos tercios de los nodos de datos tendrán que procesar tantas solicitudes como lleguen al clúster. A medida que se procesen estas solicitudes, los nodos restantes irán replicando particiones en los nuevos nodos tan pronto como se conecten, lo que podría afectar aún más al rendimiento. Si la disponibilidad es esencial para la carga de trabajo, considere la posibilidad de agregar más recursos al clúster para mitigar este problema.

> **Note**
>
> OpenSearch Service administra dominios Multi-AZ de forma transparente, por lo que no puede simular manualmente interrupciones de las zonas de disponibilidad.

Iniciar los dominios de Amazon OpenSearch Service dentro de una VPC

Puede lanzar recursos de AWS, como los dominios de Amazon OpenSearch Service, en un nube virtual privada (VPC). Una VPC es una red virtual dedicada para Cuenta de AWS. Esta infraestructura en la nube está aislada lógicamente de otras redes virtuales de la nube de AWS. Al situar un dominio de OpenSearch Service dentro de una VPC, se consigue una comunicación segura entre OpenSearch Service y los demás servicios dentro de la VPC sin necesidad de utilizar una gateway de Internet, un dispositivo NAT o una conexión de VPN. Todo el tráfico permanece seguro dentro de la nube de AWS.

> **Note**
>
> Si coloca el dominio de OpenSearch Service en una VPC, el ordenador debe poder conectarse a la VPC. Esta conexión a menudo adopta la forma de VPN, transit gateway, red administrada o servidor proxy. No puede acceder directamente a sus dominios desde fuera de la VPC.

VPC frente a dominios públicos

Las siguientes son algunas de las formas en que los dominios de VPC difieren de los dominios públicos. Cada diferencia se describe más adelante con más detalle.

- Debido a su aislamiento lógico, los dominios que residen dentro de una VPC tienen una capa adicional de seguridad en comparación con los dominios que utilizan puntos de conexión públicos.
- Aunque se puede acceder a los dominios públicos desde cualquier dispositivo conectado a Internet, los dominios de VPC requieren algún tipo de VPN o proxy.
- En comparación con los dominios públicos, los dominios VPC muestran menos información en la consola de . En concreto, la pestaña Cluster health (Estado del clúster) no incluye información de particiones y la pestaña Indices (Índices) no aparece.
- Los puntos de conexión del dominio adoptan formas diferentes (`https://search-domain-name` frente a `https://vpc-domain-name`).
- No puede aplicar estas políticas de acceso basado en IP a los dominios que residan dentro de una VPC porque los grupos de seguridad ya aplican políticas de acceso basadas en IP.

Limitaciones

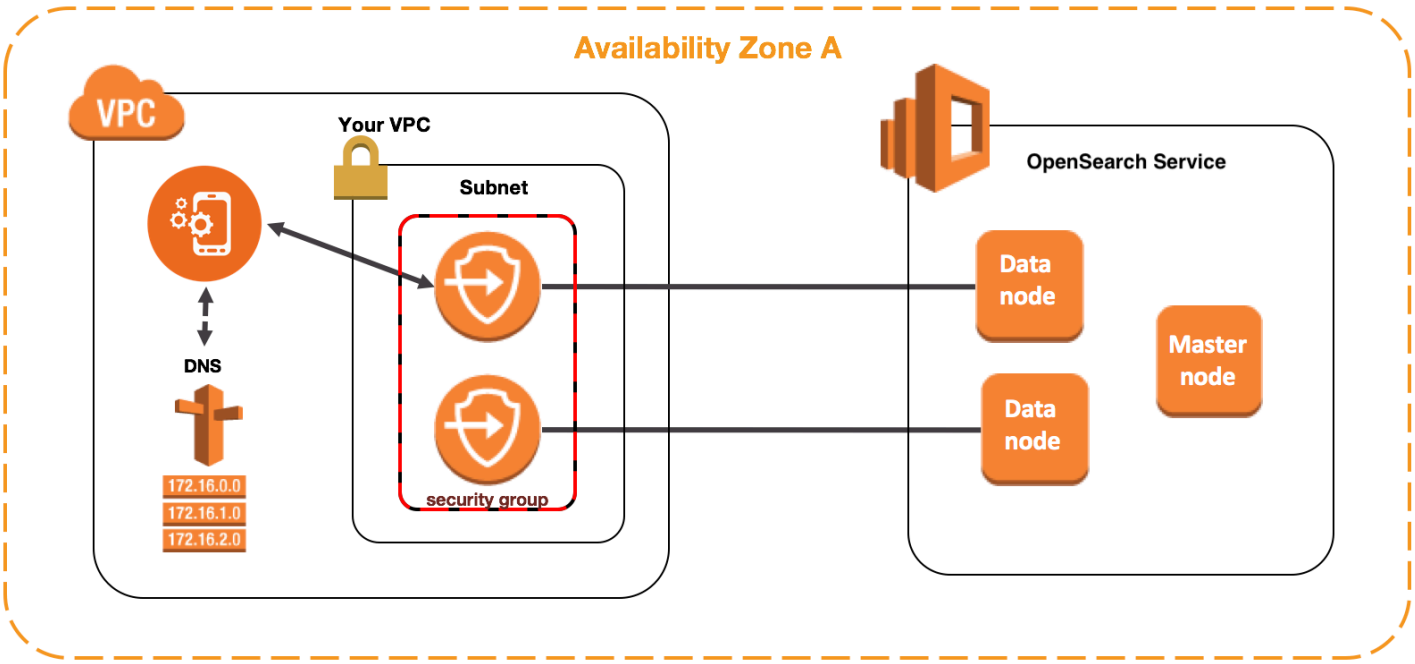
El funcionamiento de un dominio de OpenSearch Service dentro de una VPC tiene las siguientes limitaciones:

- Si se lanza un dominio en una VPC, no se puede cambiar posteriormente para que utilice un punto de conexión público. Lo contrario también es cierto: si se crea un dominio con un punto de conexión público, no es posible situarlo posteriormente en una VPC. En lugar de ello, se debe crear un dominio nuevo y migrar los datos.
- Es posible lanzar el dominio dentro de una VPC o utilizar un punto de conexión público, pero no es posible hacer las dos cosas. Se debe elegir una de ellas al crear un dominio.
- No puede lanzar el dominio en una VPC que utilice tenencia dedicada. Debe utilizar una VPC con la tenencia establecida en Default (Predeterminada).
- Después de situar un dominio dentro de una VPC, no se puede mover a otra VPC, pero puede cambiar la configuración de subredes y grupos de seguridad.
- Para tener acceso a la instalación predeterminada de OpenSearch Dashboards para un dominio que reside en una VPC, los usuarios deben tener acceso a la VPC. Este proceso depende de la configuración de la red, pero probablemente implica conectarse a una VPN o a una red gestionada, o utilizar un servidor proxy o transit gateway. Para obtener más información, consulte the section called “Acerca de las políticas de acceso en los dominios de VPC”, la Guía del usuario de Amazon VPC y the section called “Control del acceso a Dashboards ”.

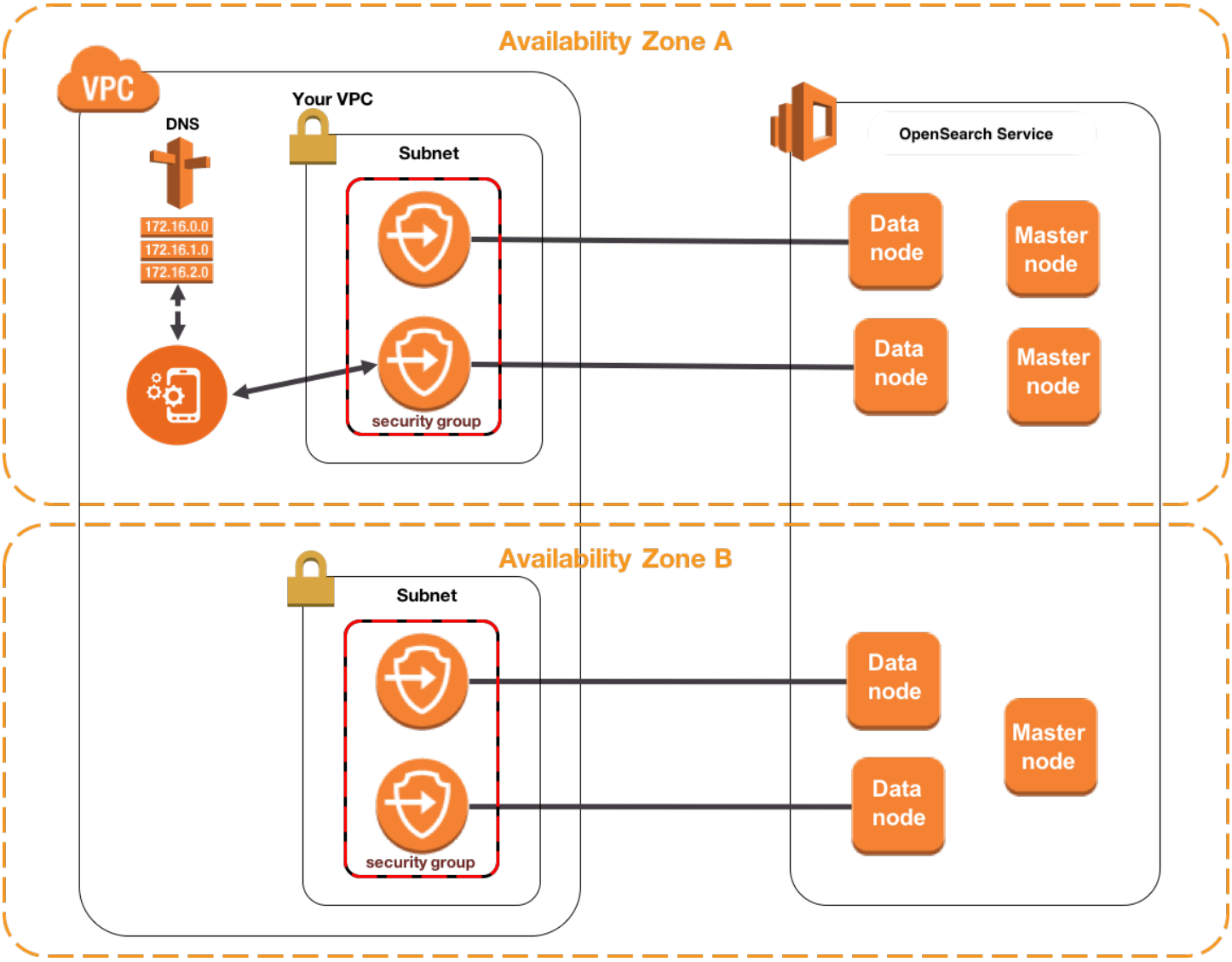
Arquitectura

Para admitir las VPC, OpenSearch Service coloca un punto de conexión en una, dos o tres subredes de la VPC. Si habilita varias zonas de disponibilidad para su dominio, cada subred debe estar en una zona de disponibilidad diferente de la misma región. Si utiliza una sola zona de disponibilidad, OpenSearch Service situará un punto de conexión en una sola subred.

En la siguiente ilustración, se muestra la arquitectura de una VPC para una zona de disponibilidad.



En la siguiente ilustración, se muestra la arquitectura de una VPC para dos zonas de disponibilidad.



OpenSearch Service también sitúa una interface de red elástica (ENI) en la VPC para cada uno de los nodos de datos. OpenSearch Service asigna a cada ENI una dirección IP privada a partir del rango de direcciones IPv4 de su subred. El servicio también asigna un nombre de host de DNS pública (que es el punto de conexión del dominio) para las direcciones IP. Es necesario utilizar un servicio DNS público para resolver el punto de conexión (que es un nombre de host DNS) a las direcciones IP adecuadas de los nodos de datos:

- Si se establece la opción `enableDnsSupport` en `true` (el valor predeterminado) para que la VPC utilice el servidor DNS proporcionado por Amazon, la resolución del punto de conexión de OpenSearch Service se realizará correctamente.

- Si la VPC utiliza un servidor DNS privado y el servidor puede conectar con servidores DNS de autorización públicos para resolver los nombres de host de DNS, la resolución del punto de conexión de OpenSearch Service también se realizará correctamente.

Dado que las direcciones IP podrían cambiar, conviene resolver periódicamente el punto de conexión del dominio para no dejar de tener acceso a los nodos de datos correctos. Recomendamos establecer el intervalo de resolución de DNS en un minuto. Si se utiliza un cliente, también conviene asegurarse de que se borra la caché de DNS del cliente.

Migración del acceso público al acceso mediante VPC

Cuando se crea un dominio, se debe especificar si debe tener un punto de conexión público o residir dentro de una VPC. Una vez creado, no se puede cambiar de un tipo a otro. Lo que hay que hacer es crear un dominio nuevo y migrar los datos o reindexarlo manualmente. Las instantáneas son un método cómodo para migrar los datos. Para obtener información sobre cómo tomar instantáneas y restaurarlas, consulte the section called “Crear instantáneas de índice”.

Acerca de las políticas de acceso en los dominios de VPC

La ubicación de un dominio de OpenSearch Service dentro de una VPC proporciona una sólida capa de seguridad inherente. Cuando crea un dominio con el acceso público, el punto de conexión adopta la siguiente forma:

```
https://search-domain-name-identifier.region.es.amazonaws.com
```

Como sugiere la etiqueta “público”, esto es punto de conexión es accesible desde cualquier dispositivo conectado a Internet, aunque puede (y debería) controlar el acceso al mismo. Si accede al punto de conexión en un navegador web, es posible que reciba un mensaje `Not Authorized`, pero la solicitud llega al dominio.

Cuando crea un dominio con acceso VPC, el punto de enlace parece similar a un punto de conexión público:

```
https://vpc-domain-name-identifier.region.es.amazonaws.com
```

Si intenta acceder al punto de conexión en un navegador web, sin embargo, podría encontrar que la solicitud agota el tiempo de espera. Para realizar solicitudes `GET` incluso básicas, su equipo debe poder conectarse a la VPC. Esta conexión a menudo adopta la forma de VPN, transit gateway, red administrada o servidor proxy. Para obtener más información sobre las distintas formas puede

adoptar, consulte Ejemplos de VPC en la Guía del usuario del Amazon VPC. Para un ejemplo de desarrollo detallado, consulte the section called “Probar los dominios de VPC”.

Además de este requisito de conectividad, las VPC le permiten administrar el acceso al dominio a través de grupos de seguridad. Para muchos casos de uso, esta combinación de características de seguridad es suficiente, y es posible que se sienta cómodo aplicando una política de acceso abierto al dominio.

El uso de una política de acceso abierto no significa que cualquier persona que se encuentre en Internet pueda tener acceso al dominio de OpenSearch Service. En su lugar, esto significa que si una solicitud llega al dominio de OpenSearch Service y los grupos de seguridad asociados la permiten, el dominio acepta la solicitud sin comprobaciones de seguridad adicionales. La única excepción es si se utiliza un control de acceso detallado o una política de acceso que especifique roles de IAM. En estas situaciones, para que el dominio acepte una solicitud, los grupos de seguridad deben permitirla y debe estar firmada con credenciales válidas.

> **Note**
>
> Dado que los grupos de seguridad ya aplican políticas de acceso basadas en IP, no puede aplicar estas políticas a los dominios de OpenSearch Service que residan dentro de una VPC. Si utiliza acceso público, las políticas basadas en IP siguen estando disponibles.

Antes de comenzar: Requisitos previos para el acceso mediante VPC

Para poder establecer una conexión entre una VPC y un dominio nuevo de OpenSearch Service, debe hacer lo siguiente:

- Cree una VPC

  Para crear una VPC, puede utilizar la consola de Amazon VPC, la CLI de AWS o uno de los SDK de AWS. Para obtener más información, consulte Uso de VPC en la Guía del usuario de Amazon VPC. Si ya tiene una VPC, puede omitir este paso.
- Reservar direcciones IP

  OpenSearch Service permite conectar una VPC a un dominio colocando interfaces de red en una subred de la VPC. Cada interfaz de red tiene asociada una dirección IP. Debe reservar un número suficiente de direcciones IP en la subred para las interfaces de red. Para obtener más información, consulte Reserva de direcciones IP en una subred de una VPC.

Probar los dominios de VPC

La seguridad mejorada de una VPC puede hacer que la conexión a su dominio y la ejecución de pruebas básicas se convierta en un reto. Si ya tiene un dominio de VPC de OpenSearch Service y prefiere no crear un servidor VPN, pruebe el siguiente proceso:

1. Para la política de acceso del dominio, elija Only use fine-grained access control (Utilizar únicamente control de acceso detallado). Siempre puede actualizar esta configuración después de finalizar las pruebas.
2. Cree una instancia de Amazon EC2 de Amazon Linux en la misma VPC, subred y grupo de seguridad que el dominio de OpenSearch Service.

   Dado que esta instancia es para fines de prueba con muy poco trabajo, elija un tipo de instancia económica como `t2.micro`. Asigne a la instancia una dirección IP pública y cree un nuevo par de claves o elija uno existente. Si crea una nueva clave, descárguela en su directorio `~/.ssh`.

   Para obtener más información acerca de la creación de instancias, consulte Introducción a las instancias de Amazon EC2 Linux.
3. Agregue un gateway de Internet a su VPC.
4. En la tabla de ruta de la VPC, agregue una nueva ruta. En Destination (Destino), especifique un bloque de CIDR que contenga la dirección IP pública de su equipo. En Target (Destino), especifique el gateway de Internet que acaba de crear.

   Por ejemplo, puede especificar solo `123.123.123.123/32` para su equipo o `123.123.123.0/24` para varios equipos.
5. Para el grupo de seguridad, especifique dos reglas de entrada:

Tipo	Protocolo	Rango de puertos	Origen
SSH (22)	TCP (6)	22	*your-cidr-block*
HTTPS (443)	TCP (6)	443	*your-security-group-id*

   La primera regla le permite usar SSH en la instancia EC2. La segunda permite que la instancia EC2 se comunique con el dominio de OpenSearch Service a través de HTTPS.
6. En el terminal ejecute el comando siguiente:

```
ssh -i ~/.ssh/your-key.pem ec2-user@your-ec2-instance-public-ip -N -L
 9200:vpc-domain-name-identifier.region.es.amazonaws.com:443
```

Este comando crea un túnel SSH que reenvía las solicitudes ahttps://localhost:9200 a su dominio de OpenSearch Service a través de la instancia EC2. La especificación del puerto 9200 en el comando simula una instalación de OpenSearch local, pero puede usar el puerto que desee. El servicio OpenSearch solo acepta conexiones a través del puerto 80 (HTTP) o 443 (HTTPS).

El comando no proporciona comentarios y se ejecuta de forma indefinida. Para detenerlo, presione `Ctrl + C`.

7. Vaya a https://localhost:9200/_dashboards/ en su navegador web. Es posible que tenga que reconocer una excepción de seguridad.

   Si lo desea, puede enviar solicitudes a https://localhost:9200 mediante curl, Postman o su lenguaje de programación favorito.

> **Tip**
>
> Si detecta errores de curl debido a que los certificados no coinciden, pruebe con la marca `--insecure`.

Reserva de direcciones IP en una subred de una VPC

OpenSearch Service conecta un dominio a una VPC, sitúa interfaces de red en una subred de la VPC (o en varias subredes de la VPC si se han habilitado varias zonas de disponibilidad). Cada interfaz de red tiene asociada una dirección IP. Antes de crear el dominio de OpenSearch Service, debe tener un número suficiente de direcciones IP disponibles en cada subred para dar cabida a las interfaces de red.

Esta es la fórmula básica: el número de direcciones IP que OpenSearch Service reserva en cada subred es tres veces el número de nodos de datos, dividido por el número de zonas de disponibilidad.

Ejemplos

- Si un dominio tiene diez nodos de datos en tres zonas de disponibilidad, el recuento de direcciones IP por cada subred será 9 * 3 / 3 = 9.

- Si un dominio tiene ocho nodos de datos en dos zonas de disponibilidad, el recuento de direcciones IP por cada subred será 8 * 3 / 2 = 12.
- Si un dominio tiene seis nodos de datos en una zona de disponibilidad, el recuento de direcciones IP por cada subred será 6 * 3 / 1 = 18.

Cuando crea el dominio, OpenSearch Service reserva las direcciones IP, usa algunas para el dominio y reserva el resto para implementaciones blue/green. Puede ver las interfaces de red y sus direcciones IP asociadas en la sección Network Interfaces (Interfaces de red) de la consola de Amazon EC2. La columna Description (Descripción) muestra el dominio de OpenSearch Service al que está asociada la interfaz de red.

> **Tip**
>
> Recomendamos que cree subredes dedicadas para las direcciones IP reservadas de OpenSearch. Mediante el uso de subredes dedicadas, evitará solapamientos con otras aplicaciones y servicios, y se asegurará de que podrá reservar direcciones IP adicionales si necesita escalar el clúster en el futuro. Para obtener más información, consulte Creación de una subred en la VPC.

También puede considerar la posibilidad de aprovisionar nodos coordinadores dedicados para reducir la cantidad de reservas de direcciones IP privadas necesarias para el dominio de VPC. OpenSearch adjunta una interfaz de red elástica (ENI) a los nodos coordinadores dedicados en lugar de adjuntarla a los nodos de datos. Los nodos coordinadores dedicados suelen representar alrededor del 10 % del total de nodos de datos. Como resultado, se reservará un número menor de direcciones IP privadas para los dominios de VPC.

Roles vinculados a servicios para el acceso mediante VPC

Un rol vinculado a un servicio es un tipo único de rol de IAM; que delega permisos en un servicio para que pueda crear y administrar recursos en nombre del usuario. OpenSearch Service necesita un rol vinculado a un servicio para tener acceso a la VPC, crear el punto de conexión del dominio y situar interfaces de red en una subred de la VPC.

OpenSearch Service crea automáticamente el rol cuando se utiliza la consola de OpenSearch Service; para crear un dominio dentro de una VPC. Para que esta creación automática se realice correctamente, es necesario disponer de permisos para la acción

`iam:CreateServiceLinkedRole`. Para obtener más información, consulte Permisos de roles vinculados a servicios en la Guía del usuario de IAM.

Después de que OpenSearch Service crea el rol, es posible verlo (`AWSServiceRoleForAmazonOpenSearchService`) utilizando la consola de IAM.

Para obtener más información acerca de los permisos de este rol y como eliminarlo, consulte the section called “Cómo utilizar roles vinculados a servicios”.

Crear instantáneas de índice en Amazon OpenSearch Service

Las instantáneas de Amazon OpenSearch Service son copias de seguridad de los índices y el estado de un clúster. Estado incluye configuraciones de clúster, información de nodos, configuración de índices y asignación de partición.

Las instantáneas de OpenSearch Service tienen las siguientes formas:

- Las instantáneas automatizadas son solo para la recuperación de clústeres. Puede utilizarlas para restaurar el dominio en caso de que se produzca un estado rojo del clúster o se pierdan datos. Para más información, consulte Restaurar instantáneas a continuación. OpenSearch Service almacena instantáneas automatizadas en un bucket de Amazon S3 preconfigurado sin costo adicional.
- Las instantáneas manuales son para la recuperación de clústeres o para el traslado de datos de un clúster a otro. Debe iniciar instantáneas manuales. Estas instantáneas se almacenan en su propio bucket de Amazon S3 y se aplican cargos estándar de S3. Si tiene una instantánea de un clúster de OpenSearch autoadministrado, puede utilizar esa instantánea para migrar a un dominio de OpenSearch Service. Para más información, consulte Migración a Amazon OpenSearch Service.

Todos los dominios de OpenSearch Service toman instantáneas automatizadas, pero la frecuencia varía de las siguientes maneras:

- En el caso de los dominios que ejecutan OpenSearch o Elasticsearch 5.3 y versiones posteriores, OpenSearch Service realiza instantáneas automáticas cada hora y retiene hasta 336 de ellas durante 14 días. Las instantáneas por hora son menos disruptivas, debido a su naturaleza progresiva. También proporcionan un punto de recuperación más reciente en caso de problemas del dominio.

- En el caso de los dominios que ejecutan Elasticsearch 5.1 y versiones anteriores, OpenSearch realiza instantáneas automatizadas diarias durante la hora que se especifique, retiene hasta 14 de ellas y no retiene ningún dato de instantáneas durante más de 30 días.

Si el clúster pasa al estado rojo, se produce un error en todas las instantáneas automatizadas mientras persiste el estado del clúster. Si no corrige el problema en dos semanas, puede perder de forma permanente los datos del clúster. Para ver los pasos de solución de problemas, consulte the section called “Estado rojo del clúster”.

Requisitos previos

Para crear instantáneas manualmente, debe trabajar con IAM y Amazon S3. Compruebe que cumple los siguientes requisitos previos antes de intentar tomar una instantánea:

Requisito previo	Descripción
Bucket de S3	Cree un bucket de S3 a fin de almacenar instantáneas manuales para su dominio de OpenSearch Service. Para obtener instrucciones, consulte Creación de un bucket de propósito general en la Guía del usuario de Amazon Simple Storage Service. Recuerde el nombre del bucket para utilizarlo en los siguientes lugares: • La instrucción `Resource` de la política de IAM que se adjunta al rol de IAM • El cliente Python utilizado para registrar un repositorio de instantáneas (si utiliza este método) **Important** No aplique una regla de ciclo de vida de Amazon Glacier a este bucket. Las instantáneas manuales no admiten la clase de almacenamiento de Amazon Glacier.
rol de IAM	Cree un rol de IAM para delegar permisos a OpenSearch Service. Para obtener instrucciones, consulte Creación de roles de IAM (consola) en la Guía del usuario de IAM. En el resto de este capítulo, este rol se denomina `TheSnapshotRole` .

Requisito previo	Descripción

	Adjuntar una política de IAM
	
	Adjunte la siguiente política a `TheSnapshotRole` para permitir el acceso al bucket de S3:
	
	`{`
	`  "Version":"2012-10-17",`
	`  "Statement": [{`
	`  "Action": [`
	`  "s3:ListBucket"`
	`  ],`
	`  "Effect": "Allow",`
	`  "Resource": [`
	`  "arn:aws:s3::: amzn-s3-demo-bucket "`
	`  ]`
	`  },`
	`  {`
	`  "Action": [`
	`  "s3:GetObject",`
	`  "s3:PutObject",`
	`  "s3:DeleteObject"`
	`  ],`
	`  "Effect": "Allow",`
	`  "Resource": [`
	`  "arn:aws:s3::: amzn-s3-demo-bucket /*"`
	`  ]`
	`  }`
	`  ]`
	`}`
	
	Para obtener instrucciones sobre cómo adjuntar una política a un rol, consulte Agregar permisos de identidad de IAM (consola) en la Guía del usuario de IAM.
	
	Modificar la relación de confianza

Requisito previo	Descripción

	Modifique la relación de confianza de `TheSnapshotRole` para especificar a OpenSearch Service en la instrucción `Principal` como se muestra en el ejemplo siguiente:
	
	`{`
	`  "Version":"2012-10-17",`
	`  "Statement": [{`
	`  "Sid": "",`
	`  "Effect": "Allow",`
	`  "Principal": {`
	`  "Service": "es.amazonaws.com"`
	`  },`
	`  "Action": "sts:AssumeRole"`
	`  }]`
	
	`}`
	
	Para obtener instrucciones a fin de editar la relación de confianza, consulte Actualización de una política de confianza de rol en la Guía del usuario de IAM.

Requisito previo	Descripción

Permisos	Para registrar el repositorio de instantáneas, necesita poder pasar `TheSnapshotRole` a OpenSearch Service. También necesita tener acceso a la acción `es:ESHttpPut`. Para conceder estos dos permisos, asocie la siguiente política al rol de IAM cuyas credenciales se utilicen para firmar la solicitud:
	
	`{ "Version":"2012-10-17", "Statement": [ { "Effect": "Allow", "Action": "iam:PassRole", "Resource": "arn:aws:iam:: 123456789012 :role/TheSnapshotRole " }, { "Effect": "Allow", "Action": "es:ESHttpPut", "Resource": "arn:aws:es: us-east-1 :123456789012 :domain/domain-name /*" } ] }`
	
	Si su usuario o rol no tiene permisos de `iam:PassRole` para transferir `TheSnapshotRole`, puede que se produzca el siguiente error común cuando intente registrar un repositorio en el siguiente paso:
	
	`$ python register-repo.py {"Message":"User: arn:aws:iam:: 123456789012 :user/MyUserAccount is not authorized to perform: iam:PassRole on resource: arn:aws:iam:: 123456789012 :role/TheSnapshotRole "}`

Registrar un repositorio de instantáneas manuales

Debe registrar el repositorio de instantáneas en OpenSearch Service para poder tomar instantáneas de índice manuales. Esta operación que solo se realiza una vez requiere que firme la solicitud de AWS con credenciales que tengan acceso a `TheSnapshotRole`, tal como se describe en the section called “Requisitos previos”.

Paso 1: asignar el rol de instantánea en OpenSearch Dashboards (si utiliza un control de acceso detallado)

El control de acceso detallado presenta un paso adicional al registrar un repositorio. Incluso si utiliza autenticación HTTP básica para todos los demás fines, debe asignar el rol `manage_snapshots` al rol de IAM que tenga permisos `iam:PassRole` para transferir `TheSnapshotRole`.

1. Desplácese hasta el complemento de OpenSearch Dashboards para ver su dominio de OpenSearch Service. Puede encontrar el punto de conexión de Dashboards en el panel del dominio de la consola de OpenSearch Service.
2. En el menú principal, seleccione Seguridad, Roles y seleccione el rol manage_snapshots.
3. Seleccione Usuarios asignados, Administrar mapeo.
4. Agregue el ARN del rol que tenga permisos para transferir `TheSnapshotRole`. Coloque los ARN de los roles en Roles de backend.

   ```
   arn:aws:iam::123456789123:role/role-name
   ```

5. Seleccione Asignar y confirme que el usuario o el rol aparecen en Usuarios asignados.

Paso 2: registrar un repositorio

La siguiente pestaña Instantáneas muestra cómo registrar un directorio de instantáneas. Para ver las opciones específicas para cifrar una instantánea manual y registrarla después de migrarla a un nuevo dominio, consulte las pestañas correspondientes.

Snapshots

Para registrar un repositorio de instantáneas, envíe una solicitud PUT al punto de conexión de dominio de OpenSearch Service. Puede usar curl, el cliente Python de muestra, Postman o algún otro método para enviar una solicitud firmada para registrar el repositorio de instantáneas. Tenga

en cuenta que no puede usar una solicitud PUT en la consola de OpenSearch Dashboards para registrar el repositorio.

La solicitud tiene el siguiente formato:

```
PUT domain-endpoint/_snapshot/my-snapshot-repo-name
{
  "type": "s3",
  "settings": {
    "bucket": "amzn-s3-demo-bucket",
    "base_path": "my/snapshot/directory",
    "region": "region",
    "role_arn": "arn:aws:iam::123456789012:role/TheSnapshotRole"
  }
}
```

> **Note**
>
> Los nombres de repositorio no pueden comenzar por “cs-”. Además, no debe escribir en el mismo repositorio desde varios dominios. Solo un dominio debe tener acceso de escritura al repositorio.

Si el dominio reside en una nube privada virtual (VPC), la computadora debe estar conectada a la VPC para que la solicitud registre correctamente el repositorio de instantáneas. El acceso a una VPC depende de la configuración de red, pero probablemente implica conectarse a una VPN o a una red corporativa. Para comprobar que tiene acceso al dominio de OpenSearch Service, diríjase a `https://your-vpc-domain.region.es.amazonaws.com` en un navegador web y compruebe que recibe la respuesta JSON predeterminada.

Cuando el bucket de Amazon S3 se encuentre en otra Región de AWS diferente del dominio de OpenSearch, añada el parámetro `"endpoint": "s3.amazonaws.com"` a la solicitud.

Encrypted snapshots

Actualmente, no puede utilizar claves AWS Key Management Service (KMS) para cifrar instantáneas manuales, pero puede protegerlas mediante el cifrado del servidor (SSE).

Para activar el SSE con claves administradas de S3 para el bucket que utiliza como repositorio de instantáneas, agregue `"server_side_encryption": true` al bloque `"settings"` de la solicitud PUT. Para obtener más información, consulte Uso del cifrado del servidor con claves

administradas por Amazon S3 (SSE-S3) en la Guía del usuario de Amazon Simple Storage Service.

También puede utilizar claves AWS KMS para el cifrado del lado del servidor en el bucket de S3 que utiliza como repositorio de instantáneas. Si utiliza este enfoque, asegúrese de proporcionar el permiso del `TheSnapshotRole` para la clave AWS KMS utilizada para cifrar el bucket de S3. Para más información, consulte Políticas de claves en AWS KMS.

Domain migration

El registro de un repositorio de instantáneas es una operación que se realiza una vez. Sin embargo, para migrar de un dominio a otro, debe registrar el mismo repositorio de instantáneas en el dominio antiguo y en el nuevo. El nombre del repositorio es arbitrario.

Tenga en cuenta las siguientes pautas al migrar a un nuevo dominio o registrar el mismo repositorio con varios dominios:

- Al registrar el repositorio en el nuevo dominio, agregue `"readonly": true` al bloque `"settings"` de la solicitud PUT. Esta configuración impide sobrescribir accidentalmente datos del dominio antiguo. Solo un dominio debe tener acceso de escritura al repositorio.
- Si migra datos a un dominio en una Región de AWS diferente (por ejemplo, desde un dominio antiguo y un bucket ubicado en us-east-2 a un nuevo dominio en us-west-2), reemplace `"region": "region"` por `"endpoint": "s3.amazonaws.com"` en la instrucción PUT y vuelva a enviar la solicitud.

Utilizar el cliente de Python de ejemplo

El cliente de Python es más fácil de automatizar que una simple solicitud HTTP y tiene una mejor reutilización. Si elige utilizar este método para registrar un repositorio de instantáneas, guarde el siguiente código de muestra de Python como archivo Python, por ejemplo `register-repo.py`. El cliente requiere los paquetes AWS SDK para Python (Boto3), requests y requests-aws4auth. El cliente contiene ejemplos de otras operaciones de instantánea que están marcados como comentarios.

Actualice las variables siguientes en el código de muestra: `host`, `region`, `path` y `payload`.

```
import boto3
import requests
from requests_aws4auth import AWS4Auth
```

```
host = '' # domain endpoint
region = '' # e.g. us-west-1
service = 'es'
credentials = boto3.Session().get_credentials()
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key, region, service,
 session_token=credentials.token)

# Register repository

path = '/_snapshot/my-snapshot-repo-name' # the OpenSearch API endpoint
url = host + path

payload = {
  "type": "s3",
  "settings": {
    "bucket": "amzn-s3-demo-bucket",
    "base_path": "my/snapshot/directory",
    "region": "us-west-1",
    "role_arn": "arn:aws:iam::123456789012:role/snapshot-role"
  }
}

headers = {"Content-Type": "application/json"}

r = requests.put(url, auth=awsauth, json=payload, headers=headers)

print(r.status_code)
print(r.text)

# # Take snapshot
#
# path = '/_snapshot/my-snapshot-repo-name/my-snapshot'
# url = host + path
#
# r = requests.put(url, auth=awsauth)
#
# print(r.text)
#
# # Delete index
#
# path = 'my-index'
# url = host + path
#
# r = requests.delete(url, auth=awsauth)
```

```
#
# print(r.text)
#
# # Restore snapshot (all indexes except Dashboards and fine-grained access control)
#
# path = '/_snapshot/my-snapshot-repo-name/my-snapshot/_restore'
# url = host + path
#
# payload = {
#   "indices": "-.kibana*,-.opendistro_security,-.opendistro-*",
#   "include_global_state": False
# }
#
# headers = {"Content-Type": "application/json"}
#
# r = requests.post(url, auth=awsauth, json=payload, headers=headers)
#
# print(r.text)
#
# # Restore snapshot (one index)
#
# path = '/_snapshot/my-snapshot-repo-name/my-snapshot/_restore'
# url = host + path
#
# payload = {"indices": "my-index"}
#
# headers = {"Content-Type": "application/json"}
#
# r = requests.post(url, auth=awsauth, json=payload, headers=headers)
#
# print(r.text)
```

Tomar instantáneas manuales

Las instantáneas no son inmediatas. Tardan un tiempo en completarse y no representan vistas perfectas de un punto en el tiempo del clúster. Mientras que una instantánea está en curso, puede indexar los documentos y realizar otras solicitudes al clúster, pero los nuevos documentos y las actualizaciones de los existentes no suelen estar incluidos en la instantánea. La instantánea incluye particiones principales que existían cuando OpenSearch inició la instantánea. En función del tamaño del grupo de subprocesos de la instantánea, se podrían incluir diferentes particiones en la instantánea a horas ligeramente diferentes. Para ver las prácticas recomendadas para instantáneas, consulte the section called “Cómo mejorar el rendimiento de las instantáneas”.

Almacenamiento y rendimiento de instantáneas

Las instantáneas de OpenSearch son progresivas, lo que significa que solo se almacenan los datos que han cambiado desde la última instantánea correcta. Esta naturaleza progresiva significa que la diferencia en la utilización de disco entre instantáneas frecuentes e infrecuentes suele ser mínima. En otras palabras, tomar instantáneas cada hora durante una semana (lo que equivale a un total de 168 instantáneas) no usaría mucho más espacio en disco que una sola instantánea al final de la semana. Además, cuanto mayor sea la frecuencia con la que toma las instantáneas, menor será el tiempo que lleve completarse. Por ejemplo, las instantáneas diarias pueden tardar entre 20 y 30 minutos en completarse, mientras que las instantáneas por hora pueden completarse en pocos minutos. Algunos usuarios de OpenSearch toman instantáneas cada media hora.

Tome una instantánea

Cuando crea una instantánea, especifica la siguiente información:

- El nombre del repositorio de instantáneas
- Un nombre de la instantánea

En los ejemplos que aparecen en este capítulo, se utiliza curl, un cliente HTTP común, por motivos de comodidad y brevedad. Para pasar un nombre de usuario y una contraseña a su solicitud de curl, consulte el tutorial de introducción.

Si las políticas de acceso especifican usuarios o roles, debe firmar las solicitudes de instantáneas. Para curl, puede usar la opción `--aws-sigv4` con la versión 7.75.0 o posterior. También puede utilizar los ejemplos comentados en el cliente Python de ejemplo para realizar solicitudes HTTP firmadas para los mismos puntos de conexión que utilizan los comandos curl.

Para realizar una instantánea manual, siga estos pasos:

1. No puede tomar una instantánea si hay alguna en curso. Para comprobarlo, ejecute el siguiente comando:

   ```
   curl -XGET 'domain-endpoint/_snapshot/_status'
   ```

2. Ejecute el siguiente comando para tomar una instantánea manual:

   ```
   curl -XPUT 'domain-endpoint/_snapshot/repository-name/snapshot-name'
   ```

Para incluir o excluir ciertos índices y especificar otras configuraciones, agregue un cuerpo de la solicitud. Para ver la estructura de solicitudes, consulte Tomar instantáneas en la documentación de OpenSearch.

> **Note**
>
> El tiempo necesario para tomar una instantánea aumenta con el tamaño del dominio de OpenSearch Service. Cuando una operación de instantánea tarda mucho tiempo en ejecutarse, en ocasiones se genera el siguiente error: `504 GATEWAY_TIMEOUT`. Normalmente, puede hacer caso omiso de estos errores y esperar a que la operación se complete correctamente. Ejecute el siguiente comando para comprobar el estado de todas las instantáneas del dominio:
>
> ```
> curl -XGET 'domain-endpoint/_snapshot/repository-name/_all?pretty'
> ```

Restaurar instantáneas

Antes de restaurar una instantánea, asegúrese de que el dominio de destino no utilice Multi-AZ con modo de espera. Si el modo de espera está activado, se produce un error en la operación de restauración.

> **Warning**
>
> Si utiliza alias de índice, debe detener las solicitudes de escritura a un alias, o cambiar el alias a otro índice, antes de borrar su índice. Al detener las solicitudes de escritura ayuda a evitar la siguiente situación:
>
> 1. Elimina un índice, de modo que también elimina su alias.
> 2. Una solicitud de escritura errante al alias que se ha eliminado crea un nuevo índice con el mismo nombre que el alias.
> 3. Ya no puede usar el alias debido a un conflicto de nombres con el nuevo índice. Si ha cambiado el alias a otro índice, especifique `"include_aliases": false` al realizar la restauración a partir de una instantánea.

Para restablecer una instantánea

1. Identifique la instantánea que desea restaurar. Asegúrese de que todos los ajustes de este índice, como los paquetes de analizadores personalizados o los ajustes de requisitos de asignación, sean compatibles con el dominio. Para ver todos los repositorios de instantáneas, ejecute el siguiente comando:

   ```
   curl -XGET 'domain-endpoint/_snapshot?pretty'
   ```

   Cuando identifique el repositorio, ejecute el siguiente comando para ver todas las instantáneas:

   ```
   curl -XGET 'domain-endpoint/_snapshot/repository-name/_all?pretty'
   ```

   > **Note**
   >
   > La mayoría de las instantáneas automatizadas se almacenan en el repositorio `cs-automated`. Si su dominio cifra datos en reposo, se almacenan en el repositorio `cs-automated-enc`. Si no ve el repositorio de instantáneas manuales que busca, asegúrese de que lo haya registrado en el dominio.

2. (Opcional) Elimine o cambie el nombre de uno o varios índices en el dominio de OpenSearch Service si hay conflictos de nomenclatura entre los índices del clúster y los índices de la instantánea. No se puede restaurar una instantánea de los índices a un clúster de OpenSearch que ya contenga índices con los mismos nombres.

   Puede seleccionar las siguientes opciones si tiene conflictos de nomenclatura de índice:

   - Elimine los índices del dominio de OpenSearch Service existente, y luego restaure la instantánea.
   - Cambie el nombre de los índices a medida que los restaure desde la instantánea y vuelva a indexarlos más tarde. Para obtener información sobre cómo cambiar el nombre de los índices, consulte esta solicitud de ejemplo en la documentación de OpenSearch.
   - Restaure la instantánea en otro dominio de OpenSearch Service (solo es posible hacerlo con las instantáneas manuales).

   El siguiente comando elimina todos los índices existentes en un dominio:

```
curl -XDELETE 'domain-endpoint/_all'
```

Sin embargo, si no piensa restaurar todos los índices, solo puede eliminar uno:

```
curl -XDELETE 'domain-endpoint/index-name'
```

3. Para restaurar una instantánea, ejecute el siguiente comando:

```
curl -XPOST 'domain-endpoint/_snapshot/repository-name/snapshot-name/_restore'
```

Debido a los permisos especiales de los índices de control de acceso detallado y los paneles de OpenSearch, es posible que los intentos de restaurar todos los índices no surtan efecto, sobre todo si se intenta restaurar desde una instantánea automatizada. En el siguiente ejemplo, se restaura un solo índice, `my-index`, a partir de `2020-snapshot` en el repositorio de instantáneas `cs-automated`:

```
curl -XPOST 'domain-endpoint/_snapshot/cs-automated/2020-snapshot/_restore' \
-d '{"indices": "my-index"}' \
-H 'Content-Type: application/json'
```

También puede que quiera restaurar todos los índices excepto los índices de control de acceso detallado y paneles:

```
curl -XPOST 'domain-endpoint/_snapshot/cs-automated/2020-snapshot/_restore' \
-d '{"indices": "-.kibana*,-.opendistro*"}' \
-H 'Content-Type: application/json'
```

Puede restaurar una instantánea sin eliminar sus datos mediante los parámetros `rename_pattern` y `rename_replacement`. Para obtener más información sobre estos parámetros, consulte los campos de solicitud de la API Restore Snapshot y la solicitud de ejemplo en la documentación de OpenSearch.

> **Note**
>
> Si no están disponibles todas las particiones principales para los índices implicados, es posible que una instantánea tenga `state` establecido como `PARTIAL`. Este valor indica que

los datos de al menos una partición no se han almacenado correctamente. Puede continuar con la restauración a partir de una instantánea parcial, pero es posible que tenga que utilizar instantáneas anteriores para restaurar los índices que falten.

Eliminar instantáneas manuales

Ejecute el siguiente comando para tomar una instantánea manual:

```
DELETE _snapshot/repository-name/snapshot-name
```

Automatizar instantáneas con la administración de instantáneas

Puede configurar una política de administración de instantáneas (SM) en los OpenSearch Dashboards para automatizar la creación y eliminación periódicas de instantáneas. SM puede tomar una instantánea de un grupo de índices, mientras que Administración de estados de índice solo puede tomar una instantánea por índice. Para usar SM en OpenSearch Service, debe registrar su propio repositorio de Amazon S3. Para ver instrucciones sobre cómo registrar su repositorio, consulte Registrar un repositorio de instantáneas manuales.

Antes de SM, OpenSearch Service ofrecía una característica de instantáneas automática y gratuita que todavía estaba activada de forma predeterminada. Esta característica envía las instantáneas al repositorio mantenido por el servicio`cs-*`. Para desactivar la característica, póngase en contacto con Soporte.

Para ver más información sobre la característica SM, consulte Administración de instantáneas en la documentación de OpenSearch.

Actualmente, SM no admite la creación de instantáneas en varios tipos de índices. Por ejemplo, si intenta crear una instantánea en varios índices con * y algunos índices están en el nivel semiactivo, la creación de la instantánea producirá un error. Si necesita que la instantánea contenga varios tipos de índices, utilice la acción de instantánea de ISM hasta que SM admita esta opción.

Configuración de permisos

Si está actualizando a la versión 2.5 desde una versión de dominio de OpenSearch Service anterior, los permisos de seguridad de administración de instantáneas podrían no estar definidos en el dominio. Los usuarios que no sean administradores deben estar asignados a este rol para usar la

administración de instantáneas en los dominios mediante un control de acceso detallado. Para crear el rol de administración de instantáneas de forma manual, siga estos pasos:

1. En OpenSearch Dashboards, vaya a Seguridad y elija Permisos.
2. Seleccione Crear grupo de acciones y configure los siguientes grupos:

Nombre del grupo	Permisos
`snapshot_management_full_access`	• `cluster:admin/opensearch/snapshot_management/*` • `cluster:admin/opensearch/notifications/feature/publish` • `cluster:admin/repository/*` • `cluster:admin/snapshot/*`
`snapshot_management_read_access`	• `cluster:admin/opensearch/snapshot_management/policy/get` • `cluster:admin/opensearch/snapshot_management/policy/search` • `cluster:admin/opensearch/snapshot_management/policy/explain` • `cluster:admin/repository/get` • `cluster:admin/snapshot/get`

3. Seleccione Roles y, a continuación, Crear rol.
4. Asigne el nombre snapshot_management_role al rol.
5. Para Permisos de clúster, seleccione `snapshot_management_full_access` o `snapshot_management_read_access`.
6. Seleccione Crear.
7. Después de crear el rol, debe mapearlo a cualquier rol de usuario o backend que administre instantáneas.

Consideraciones

Tenga en cuenta lo siguiente al configurar la administración de instantáneas:

- Se permite una política por repositorio.
- Se permiten hasta 400 instantáneas para una política.
- Esta característica no se ejecutará si su dominio tiene un estado rojo, está sometido a una presión elevada por la JVM (un 85 % o más) o tiene una característica de captura bloqueada. Si el rendimiento general de indexación y búsqueda de su clúster se ve afectado, es posible que SM también se vea afectado.
- Una operación de instantánea solo comienza una vez finalizada la operación anterior, por lo que una política no activa ninguna operación de instantánea simultánea.
- Varias políticas con la misma programación pueden provocar un pico de recursos. Si los índices instantáneos de las políticas se superponen, las operaciones de instantáneas a nivel de partición solo se pueden ejecutar de forma secuencial, lo que puede provocar un problema de rendimiento en cascada. Si las políticas comparten un repositorio, habrá un pico de operaciones de escritura en ese repositorio.
- Le recomendamos que programe la automatización de sus operaciones instantáneas para que no se realice más de una vez por hora, a menos que se trate de un caso de uso especial.

Automatizar instantáneas con la administración de estado de índice

Puede utilizar la operación de administración de estado de índice (ISM) para activar automáticamente instantáneas de índices en función de los cambios en su antigüedad, tamaño o número de documentos. ISM es la mejor opción cuando se necesita una instantánea por índice. Si necesita hacer una instantánea de un grupo de índices, consulte Automatizar instantáneas con la administración de instantáneas.

Para usar SM en OpenSearch Service, debe registrar su propio repositorio de Amazon S3. Para obtener un ejemplo de política ISM con la operación `snapshot`, consulte Políticas de muestra.

Utilizar Curator para instantáneas

Si ISM no funciona para la administración de índices e instantáneas, puede utilizar Curator en su lugar. Ofrece una funcionalidad avanzada de filtrado que ayuda a simplificar tareas de administración en clústeres complejos. Utilice pip para instalar Curator:

```
pip install elasticsearch-curator
```

Puede utilizar Curator como una interfaz de línea de comandos (CLI) o una API de Python. Si utiliza la API de Python, debe utilizar la versión 7.13.4 o anterior del cliente elasticsearch-py heredado. No admite el cliente opensearch-py.

Si utiliza la CLI, exporte sus credenciales en la línea de comandos y configure `curator.yml` como se indica a continuación:

```
client:
  hosts: search-my-domain.us-west-1.es.amazonaws.com
  port: 443
  use_ssl: True
  aws_region: us-west-1
  aws_sign_request: True
  ssl_no_validate: False
  timeout: 60

logging:
  loglevel: INFO
```

Actualización de los dominios de Amazon OpenSearch Service

> **Note**
>
> Las actualizaciones de la versión de Elasticsearch se diferencian de las actualizaciones del software de servicio. Para obtener información acerca de cómo actualizar el software del servicio de su dominio de OpenSearch Service, consulte the section called “Actualizaciones del software de servicio”.

Amazon OpenSearch Service ofrece actualizaciones de locales para los dominios que ejecutan OpenSearch 1.0 o posterior, o Elasticsearch 5.1 o posterior. Si utiliza servicios como Amazon Data Firehose o Registros de Amazon CloudWatch para transmitir datos a OpenSearch Service, compruebe que estos servicios admitan la versión más reciente de OpenSearch antes de la migración.

Rutas de actualización admitidas

OpenSearch Service admite actualmente las siguientes rutas de actualización:

Desde la versión	Hasta la versión
OpenSearch 1.3 o 2.x	OpenSearch 3.1 Si está ejecutando OpenSearch 1.3 o 2.x, primero debe actualizar a OpenSearch 2.19 antes de actualizar a OpenSearch 3.1. Posteriormente, cuando actualice de OpenSearch 2.19 a OpenSearch 3.1, es posible que tenga que corregir las siguientes configuraciones de índice incompatibles. Quedaron en desuso a principios de la versión 2.x y, si no se eliminan, provocarán un error en la validación de la comprobación de la actualización. • index.knn.algo_param.ef_construction • index.knn.algo_param.m • index.knn.space_type • index.store.hybrid.mmap.extensions Para obtener más información sobre la configuración de índice obsoleta y los pasos para corregirla, consulte Configuración de índices. Para obtener una lista completa de los cambios más importantes de OpenSearch 3.1, consulte la sección 3.0.0.
OpenSearch 1.3 o 2.x	OpenSearch 2.x OpenSearch 2.17 habilitará la búsqueda simultánea de segmentos de forma predeterminada con el modo automático si el dominio cumple las siguientes condiciones: • No se ha establecido explícitamente ninguna configuración de búsqueda simultánea previa. • Todas las instancias de datos (calientes y cálidos) son del tipo 2.xl o superior. • El uso medio de la CPU p90 en las instancias de datos (calientes y cálidos) durante más de una semana es inferior al 45 %. Para obtener más información sobre la configuración de búsqueda de segmentos simultáneos aquí, consulte Búsqueda de segmentos simultáneos.

Desde la versión	Hasta la versión
	El control de versiones 2.3 incluye los siguientes cambios importantes: • Este parámetro `type` se eliminó de todos los puntos de conexión de la API OpenSearch en el control de versiones 2.0. Para más información, consulte Cambios bruscos. • Si su dominio contiene algún índice (almacenado en caliente, UltraWarm o frío) que se haya creado originalmente en Elasticsearch 6.8, esos índices no son compatibles con OpenSearch 2.3. Antes de actualizar a la versión 2.3, debe volver a indexar los índices incompatibles. En el caso de índices UltraWarm o frío incompatibles, migre a un almacenamiento en caliente, vuelva a indexar los datos y, a continuación, vuelva a migrarlos a un almacenamiento en caliente o frío. También puede quitar índices si ya no los necesita. Si actualiza accidentalmente su dominio al control de versiones 2.3 sin realizar primero estos pasos, no podrá migrar los índices incompatibles fuera de su nivel de almacenamiento actual. La única opción es eliminarlos.
OpenSearch 1.x	OpenSearch 1.x
Elasticsearch 7.x	Elasticsearch 7.x u OpenSearch 1.x **Important** OpenSearch 1.x presenta muchos cambios importantes. Para más información, consulte Cambio de nombre de Amazon OpenSearch Service.

Desde la versión	Hasta la versión
Elasticsearch 6.8	Elasticsearch 7.x u OpenSearch 1.x **Important** Elasticsearch 7.0 y OpenSearch 1.0 incluyen numerosos cambios de ruptura. Antes de iniciar una actualización in situ, se recomienda tomar una instantánea manual del dominio 6.x, restaurarla en un dominio 7.x de prueba o de OpenSearch 1.x y utilizar ese dominio de prueba para identificar posibles problemas de actualización. Para ver los cambios importantes en OpenSearch 1.0, consulte Cambio de nombre de Amazon OpenSearch Service. Al igual que Elasticsearch 6.x, los índices solo pueden contener un tipo de mapeo, pero ese tipo debe denominarse `_doc`. Como resultado, determinadas API ya no requieren un tipo de mapeo en el cuerpo de la solicitud (como, por ejemplo, la API `_bulk`). Para nuevos índices, Elasticsearch 7.x y OpenSearch 1.x de alojamien to propio tienen un recuento de particiones predeterminado de uno. Los dominios de OpenSearch Service en Elasticsearch 7.x y posteriores conservan el valor predeterminado anterior de cinco.
Elasticsearch 6.x	Elasticsearch 6.x

Desde la versión	Hasta la versión
Elasticsearch 5.6	Elasticsearch 6.x **Important** Los índices creados en la versión 6.x ya no admiten varios tipos de mapeo. Los índices creados en la versión 5.x siguen admitiendo varios tipos de mapeo cuando se restauran en un clúster 6.x. Compruebe que el código de cliente solo cree un único tipo de mapeo por índice. Para minimizar el tiempo de inactividad durante el actualización de Elasticsearch 5.6 a 6.x, OpenSearch Service reindexa el índice `.kibana` a `.kibana-6` , elimina `.kibana`, crea un alias denominado `.kibana` y asigna el nuevo índice al nuevo alias.
Elasticsearch 5.x	Elasticsearch 5.x

El proceso de actualización consta de tres pasos:

1. Comprobaciones de preactualizuación: OpenSearch Service realiza comprobaciones para detectar problemas que puedan bloquear una actualización y evitar continuar con el siguiente paso a menos que estas comprobaciones sean satisfactorias.
2. Instantánea: OpenSearch Service toma una instantánea del clúster de OpenSearch o Elasticsearch y no continúa con el siguiente paso a menos que la instantánea se realice satisfactoriamente. Si se produce un error en la actualización, OpenSearch Service utiliza esta instantánea para restaurar el clúster a su estado original. Para más información, consulte the section called “No se puede cambiar a una versión anterior después de una actualización”.
3. Actualización: OpenSearch Service comienza la actualización, lo cual puede tardar en completarse de 15 minutos a varias horas. OpenSearch Dashboards podría no estar disponible durante parte o la totalidad de la actualización.

Actualización de un dominio (consola)

El proceso de actualización es irreversible y no se puede pausar ni cancelar. Durante una actualización no puede realizar cambios de configuración en el dominio. Antes de comenzar una actualización, compruebe que desea continuar. Puede utilizar estos mismos pasos para realizar la verificación de actualización sin iniciar realmente una actualización.

Si el clúster tiene nodos maestros dedicados, las actualizaciones de OpenSearch se completan sin tiempo de inactividad. De lo contrario, el clúster podría no responder durante varios segundos después de la actualización mientras elige un nodo maestro.

Para actualizar un dominio a una versión posterior de OpenSearch o Elasticsearch

1. Tome una instantánea manual de su dominio. Esta instantánea sirve como copia de seguridad que puede restaurar en un nuevo dominio si desea volver a utilizar la versión de OpenSearch anterior.
2. Visite https://aws.amazon.com y elija Iniciar sesión en la consola.
3. En Análisis, seleccione Amazon OpenSearch Service.
4. En el panel de navegación, en Dominios, seleccione el dominio que desea actualizar.
5. Seleccione Acciones y Actualizar.
6. Seleccione la version a la que actualizar. Si está actualizando a una versión de OpenSearch, aparece la opción Habilitar modo de compatibilidad. Si habilita esta configuración, OpenSearch informa de su versión como 7.10 para permitir que los clientes y complementos de Elasticsearch OSS como Logstash continúen trabajando con Amazon OpenSearch Service. Podrá desactivar esta configuración más tarde
7. Seleccione Actualizar.
8. Verifique el Estado en el panel del dominio para monitorear el estado de la actualización.

Actualización de un dominio (CLI)

Puede utilizar las siguientes operaciones para identificar la versión de OpenSearch o Elasticsearch correcta para su dominio, iniciar una actualización local, realice la comprobación previa a la actualización y ver el progreso:

- `get-compatible-versions` (`GetCompatibleVersions`)
- `upgrade-domain` (`UpgradeDomain`)

- `get-upgrade-status` (`GetUpgradeStatus`)
- `get-upgrade-history` (`GetUpgradeHistory`)

Para más información, consulte AWS CLI Command Reference y Amazon OpenSearch Service API Reference.

Actualización de un dominio (SDK)

En esta muestra, se utiliza el cliente OpenSearchService de Python de bajo nivel desde AWS SDK for Python (Boto) para comprobar si un dominio es apto para actualizarse a una versión específica, actualizarlo y comprobar continuamente el estado de la actualización.

```
import boto3
from botocore.config import Config
import time

# Build the client using the default credential configuration.
# You can use the CLI and run 'aws configure' to set access key, secret
# key, and default Region.

DOMAIN_NAME = ''  # The name of the domain to upgrade
TARGET_VERSION = ''  # The version you want to upgrade the domain to. For example,
 OpenSearch_1.1

my_config = Config(
    # Optionally lets you specify a Region other than your default.
    region_name='us-east-1'
)
client = boto3.client('opensearch', config=my_config)


def check_versions():
    """Determine whether domain is eligible for upgrade"""
    response = client.get_compatible_versions(
        DomainName=DOMAIN_NAME
    )
    compatible_versions = response['CompatibleVersions']
    for i in range(len(compatible_versions)):
        if TARGET_VERSION in compatible_versions[i]["TargetVersions"]:
            print('Domain is eligible for upgrade to ' + TARGET_VERSION)
            upgrade_domain()
            print(response)
```

```
        else:
            print('Domain not eligible for upgrade to ' + TARGET_VERSION)


def upgrade_domain():
    """Upgrades the domain"""
    response = client.upgrade_domain(
        DomainName=DOMAIN_NAME,
        TargetVersion=TARGET_VERSION
    )
    print('Upgrading domain to ' + TARGET_VERSION + '...' + response)
    time.sleep(5)
    wait_for_upgrade()


def wait_for_upgrade():
    """Get the status of the upgrade"""
    response = client.get_upgrade_status(
        DomainName=DOMAIN_NAME
    )
    if (response['UpgradeStep']) == 'UPGRADE' and (response['StepStatus']) ==
 'SUCCEEDED':
        print('Domain successfully upgraded to ' + TARGET_VERSION)
    elif (response['StepStatus']) == 'FAILED':
        print('Upgrade failed. Please try again.')
    elif (response['StepStatus']) == 'SUCCEEDED_WITH_ISSUES':
        print('Upgrade succeeded with issues')
    elif (response['StepStatus']) == 'IN_PROGRESS':
        time.sleep(30)
        wait_for_upgrade()


def main():
    check_versions()


if __name__ == "__main__":
    main()
```

Solución de errores de validación

Al iniciar una de versión de OpenSearch o Elasticsearch, OpenSearch Service primero lleva a cabo una serie de comprobaciones de validación para garantizar que el dominio es apto para una

actualización. Si se produce un error en alguna de estas comprobaciones, recibirá una notificación con los problemas específicos que debe corregir antes de actualizar su dominio. Para obtener una lista de posibles problemas y los pasos para resolverlos, consulte the section called “Solución de errores de validación”.

Solución de problemas de una actualización

Las actualizaciones locales requieren dominios en buen estado. Es posible que su dominio no pueda optar por una actualización o que no se actualice por diversas razones. La siguiente tabla muestra los problemas más comunes.

Problema	Descripción
No se admite el complemento opcional	Cuando actualiza un dominio con complementos opcionales, OpenSearch Service también actualiza automáticamente los complementos. Por lo tanto, la versión de destino de su dominio también debe admitir estos complementos opcionales. Si el dominio tiene instalado un complemento opcional que no está disponible para la versión de destino, se produce un error en la solicitud de actualización.
Demasiadas particiones por nodo	OpenSearch y las versiones 7.x de Elasticsearch tienen una configuración predeterminada de no más de 1000 particiones por nodo. Si un nodo del clúster actual supera esta configuración, OpenSearch Service no le permite actualizar. Consulte the section called “Límite máximo de fragmentos superado” para obtener opciones de solución de problemas.
Dominio en procesamiento	El dominio está en medio de un cambio de configuración. Compruebe si es posible optar a la actualización una vez que se complete la operación.
Estado rojo del clúster	Uno o varios índices en el clúster aparecen en rojo. Para ver los pasos de solución de problemas, consulte the section called “Estado rojo del clúster”.
Tasa de errores alta	El clúster devuelve un gran número de errores 5xx al intentar procesar solicitudes. Este problema suele ser el resultado de demasiadas solicitudes de lectura o escritura simultáneas. Considere la posibilidad de reducir el tráfico hacia el clúster o el escalado del dominio.

Problema	Descripción
Cerebro dividido	Cerebro dividido significa que el clúster tiene más de un nodo maestro y se ha dividido en dos clústeres que nunca se volverán a unir por sí mismos. Puede evitar un cerebro dividido con el número recomendado de nodos maestros dedicados. Para obtener ayuda a fin de recuperar un cerebro dividido, póngase en contacto con Soporte.
Nodo maestro no encontrado	OpenSearch Service no puede encontrar el nodo maestro del clúster. Si el dominio utiliza Multi-AZ, es posible que un error de una zona de disponibilidad haya hecho que el clúster pierda el quorum y no pueda elegir un nuevo nodo maestro. Si el problema no se resuelve por sí solo, póngase en contacto con Soporte.
Demasiadas tareas pendientes	El nodo maestro está bajo una gran carga y tiene muchas tareas pendientes. Considere la posibilidad de reducir el tráfico hacia el clúster o el escalado del dominio.
Volumen de almacenamiento dañado	El volumen de disco de uno o más nodos no funciona correctamente. Este problema a menudo ocurre junto con otros problemas, como una alta tasa de error o demasiadas tareas pendientes. Si se produce aisladamente y no se resuelve por sí solo, póngase en contacto con Soporte.
Problema de clave de KMS	La clave de KMS que se utiliza para cifrar el dominio está inaccesible o no está. Para más información, consulte the section called “Monitorear dominios que cifran los datos en reposo”.
Snapshot en proceso de creación	El dominio está tomando una instantánea en este momento. Compruebe si es posible optar a la actualización una vez que se complete la instantánea. Compruebe también si puede hacer una lista de los repositorios de instantáneas manuales, una lista de instantáneas dentro de dichos repositorios y tomar instantáneas manuales. Si OpenSearch Service no puede comprobar si se está realizando una instantánea, las actualizaciones puede generar errores.

Problema	Descripción
Error en instantánea o tiempo de espera agotado	La instantánea previa a la actualización ha tardado demasiado tiempo en completarse o ha generado un error. Compruebe el estado del clúster y vuelva a intentarlo. Si el problema sigue sin resolverse, póngase en contacto con Soporte.
Índices incompatibles	Uno o varios índices son incompatibles con la versión de destino. Este problema puede ocurrir si migró los índices desde una versión anterior de OpenSearch o Elasticsearch. Reindexe los índices e inténtelo de nuevo.
Alto uso del disco	El uso del disco para el clúster es superior al 90 %. Elimine los datos o escale el dominio, e inténtelo de nuevo.
Alto uso de JVM	La presión de memoria de JVM es superior al 75 %. Reduzca el tráfico hacia el clúster o amplíe el dominio, e inténtelo de nuevo.
Problema de alias de OpenSearch Dashboards	`.dashboards` ya está configurado como un alias y se asigna a un índice incompatible, probablemente uno de una versión anterior de OpenSearch Dashboards. Reindexe e inténtelo de nuevo.
Estado rojo de Dashboards	El estado de OpenSearch Dashboards es rojo. Intente utilizar Dashboard s cuando se complete la actualización. Si el estado rojo persiste, resuélvalo manualmente, e inténtelo de nuevo.
Compatibilidad entre clústeres	Solo puede actualizar si se mantiene la compatibilidad entre clústeres entre los dominios de origen y destino después de la actualización. Durante el proceso de actualización, se identifican las conexiones incompatibles. Para continuar, actualice el dominio remoto o elimine las conexiones incompatibles. Tenga en cuenta que si la replicación está activa en el dominio, no podrá reanudarla una vez que elimine la conexión.
Otro problema de servicio de OpenSearch Service	Los problemas con OpenSearch Service en sí pueden hacer que el dominio no pueda optar a una actualización. Si a su dominio no se aplica ninguna de las condiciones anteriores y el problema persiste durante más de un día, póngase en contacto con Soporte.

Uso de una instantánea para migrar datos

Las actualizaciones locales son más fáciles, más rápidas y la forma más fiable de actualizar un dominio a una versión de OpenSearch o Elasticsearch posterior. Las instantáneas son una buena opción si tiene que migrar de una versión previa a la 5.1 de Elasticsearch o si desea migrar a un clúster totalmente nuevo.

En la siguiente tabla se muestra cómo utilizar instantáneas para migrar datos a un dominio que utiliza una versión de OpenSearch o Elasticsearch diferente. Para obtener información sobre cómo tomar instantáneas y restaurarlas, consulte the section called “Crear instantáneas de índice”.

Desde la versión	Hasta la versión	Proceso de migración

OpenSearch 1.3 o 2.x	OpenSearch 2.x	1. Consulte los cambios importantes en OpenSearch 2.3 para saber si necesita realizar ajustes en los índices o en la aplicación.
		2. Cree una instantánea manualmente del dominio 1.3 o 2.x.
		3. Cree un dominio 2.x que sea una versión superior al dominio 1.3 o 2.x original.
		4. Restaure la instantánea desde el dominio original al dominio 2.x. Durante la operación, es probable que tenga que restaurar el índice `.opensearch` con un nuevo nombre:
		`POST _snapshot/ <repository-name> /<snapshot-name>/_restore`
		`{`
		`  "indices": "*",`
		`  "ignore_unavailable": true,`
		`  "rename_pattern": ".opensearch",`
		`  "rename_replacement": ".backup-opensearch"`
		`}`
		A continuación, puede reindexar `.backup-opensearch` en el nuevo dominio y asignarlo a `.opensearch`. Tenga en cuenta que la llamada

Desde la versión	Hasta la versión	Proceso de migración
		REST `_restore` no incluye `include_global_state` porque la entrada predeterminada en `_restore` es falsa. Como resultado, el dominio de prueba no incluirá ninguna plantilla de índice y no tendrá el estado completo de la copia de seguridad. 5. Si ya no necesita el dominio original, elimínelo. De lo contrario, el dominio seguirá generando costos.

Desde la versión	Hasta la versión	Proceso de migración

OpenSearch 1.x	OpenSearch 1.x	1. Cree una instantánea manual del dominio 1.x.
		2. Cree un dominio 1.x que sea una versión superior al dominio 1.x original.
		3. Restaure la instantánea desde el dominio original al nuevo dominio 1.x. Durante la operación, es probable que tenga que restaurar el índice `.opensearch` con un nuevo nombre:
		`POST _snapshot/ <repository-name> /<snapshot-name>/_restore`
		`{`
		`  "indices": "*",`
		`  "ignore_unavailable": true,`
		`  "rename_pattern": ".opensearch",`
		`  "rename_replacement": ".backup-opensearch"`
		`}`
		A continuación, puede reindexar `.backup-opensearch` en el nuevo dominio y asignarlo a `.opensearch`. Tenga en cuenta que la llamada REST `_restore` no incluye `include_global_state` porque la entrada predeterminada en `_restore` es falsa. Como resultado, el dominio de prueba no incluirá ninguna plantilla de índice y no tendrá el estado completo de la copia de seguridad.
		4. Si ya no necesita el dominio original, elimínelo. De lo contrario, el dominio seguirá generando costos.

Desde la versión	Hasta la versión	Proceso de migración

Elasticsearch 6.x o 7.x	OpenSearch 1.x	1. Consulte los cambios importantes en OpenSearch 1.0 para saber si necesita realizar ajustes en los índices o en la aplicación.
		2. Cree una instantánea manualmente del dominio de Elasticsearch 7.x o 6.x.
		3. Cree un dominio de OpenSearch 1.x.
		4. Restaure la instantánea desde el dominio Elasticsearch al dominio de OpenSearch. Durante la operación, es probable que tenga que restaurar el índice `.elasticsearch` con un nuevo nombre:
		`POST _snapshot/ <repository-name> /<snapshot-name>/_restore`
		`{`
		`  "indices": "*",`
		`  "ignore_unavailable": true,`
		`  "rename_pattern": ".elasticsearch",`
		`  "rename_replacement": ".backup-opensearch"`
		`}`
		A continuación, puede reindexar `.backup-opensearch` en el nuevo dominio y asignarlo a `.elasticsearch`. Tenga en cuenta que la llamada REST `_restore` no incluye `include_global_state` porque la entrada predeterminada en `_restore` es falsa. Como resultado, el dominio de prueba no incluirá ninguna plantilla de índice y no tendrá el estado completo de la copia de seguridad.
		5. Si ya no necesita el dominio original, elimínelo. De lo contrario, el dominio seguirá generando costos.

Desde la versión	Hasta la versión	Proceso de migración

Elasticsearch 6.x	Elasticsearch 7.x	1. Consulte los cambios importantes en 7.0 para saber si necesita realizar ajustes en los índices o en la aplicación.
		2. Cree una instantánea manual del dominio 6.x.
		3. Cree un dominio de la versión 7x.
		4. Restaure la instantánea desde el dominio original al dominio 7x. Durante la operación, es probable que tenga que restaurar el índice `.opensearch` con un nuevo nombre:
		`POST _snapshot/ <repository-name> /<snapshot-name>/_restore`
		`{`
		`  "indices": "*",`
		`  "ignore_unavailable": true,`
		`  "rename_pattern": ".elasticsearch",`
		`  "rename_replacement": ".backup-elasticsearch"`
		`}`
		A continuación, puede reindexar `.backup-elasticsearch` en el nuevo dominio y asignarlo a `.elasticsearch` . Tenga en cuenta que la llamada REST `_restore` no incluye `include_global_state` porque la entrada predeterminada en `_restore` es falsa. Como resultado, el dominio de prueba no incluirá ninguna plantilla de índice y no tendrá el estado completo de la copia de seguridad.
		5. Si ya no necesita el dominio original, elimínelo. De lo contrario, el dominio seguirá generando costos.

Desde la versión	Hasta la versión	Proceso de migración
Elasticsearch 6.x	Elasticsearch 6.8	1. Cree una instantánea manual del dominio 6.x. 2. Cree un dominio de la versión 6.8. 3. Restaure la instantánea desde el dominio original al dominio 6.8. 4. Si ya no necesita el dominio original, elimínelo. De lo contrario, el dominio seguirá generando costos.
Elasticsearch 5.x	Elasticsearch 6.x	1. Consulte los cambios importantes en 6.0 para saber si necesita realizar ajustes en los índices o en la aplicación. 2. Cree una instantánea manual del dominio 5.x. 3. Cree un dominio de la versión 6.x. 4. Restaure la instantánea desde el dominio original al dominio 6.x. 5. Si ya no necesita el dominio 5.x, elimínelo. De lo contrario, el dominio seguirá generando costos.
Elasticsearch 5.x	Elasticsearch 5.6	1. Cree una instantánea manual del dominio 5.x. 2. Cree un dominio de la versión 5.6. 3. Restaure la instantánea desde el dominio original al dominio 5.6. 4. Si ya no necesita el dominio original, elimínelo. De lo contrario, el dominio seguirá generando costos.

Desde la versión	Hasta la versión	Proceso de migración
Elasticsearch 2.3	Elasticsearch 6.x	Las instantáneas de Elasticsearch 2.3 no son compatibles con la versión 6.x. Para migrar sus datos directamente de la versión 2.3 a 6.x, debe volver a crear manualmente sus índices en el nuevo dominio. Como alternativa, puede seguir los pasos del cambio de la versión 2.3 a 5.x de esta tabla, realizar las operaciones de `_reindex` en el nuevo dominio 5.x para convertir sus índices 2.3 en índices 5.x y después seguir los pasos del cambio de la versión 5.x a 6.x.
Elasticsearch 2.3	Elasticsearch 5.x	1. Consulte los cambios importantes en 5.0 para saber si necesita realizar ajustes en los índices o en la aplicación. 2. Cree una instantánea manualmente del dominio 2.3. 3. Cree un dominio de la versión 5.x. 4. Restaure la instantánea desde el dominio 2.3 al dominio 5.x. 5. Si ya no necesita el dominio 2.3, elimínelo. De lo contrario, el dominio seguirá generando costos.
Elasticsearch 1.5	Elasticsearch 5.x	Las instantáneas de Elasticsearch 1.5 no son compatibles con la versión 5.x. Para migrar sus datos de la versión 1.5 a 5.x, debe volver a crear manualmente sus índices en el nuevo dominio. **Important** Las instantáneas de 1.5 son compatibles con los dominios 2.3, pero los dominios de OpenSearch Service 2.3 no admiten la operación `_reindex`. Dado que no puede reindexarlos, los índices que se originaron en un dominio 1.5 siguen sin restaurar instantáneas de 2.3 a dominios 5x.

Desde la versión	Hasta la versión	Proceso de migración
Elasticsearch 1.5	Elasticsearch 2.3	1. Utilice el complemento de migración para saber si puede hacer directamente la actualización a la versión 2.3. Es posible que tenga que realizar cambios en los datos antes de la migración. a. En un navegador web, abra `http://`*`domain-endpoint`*` /_plugin/migration/` . b. Seleccione Realizar comprobaciones ahora. c. Revise los resultados y, si es necesario, siga las instrucciones para realizar cambios en los datos. 2. Cree una instantánea manualmente del dominio 1.5. 3. Cree un dominio de la versión 2.3. 4. Restaure la instantánea desde el dominio 1.5 al dominio 2.3. 5. Si ya no necesita el dominio 1.5, elimínelo. De lo contrario, el dominio seguirá generando costos.

Creación de un punto de conexión personalizado para Amazon OpenSearch Service

La creación de un punto de conexión personalizado para un dominio de Amazon OpenSearch Service facilita la consulta de las direcciones URL de OpenSearch y de OpenSearch Dashboards. Puede incluir la marca de su empresa o simplemente utilizar un punto de conexión más corto y fácil de recordar que el estándar.

Si alguna vez necesita cambiar a un nuevo dominio, simplemente actualice el DNS para que apunte a la nueva URL y continúe utilizando el mismo punto de conexión anterior.

Proteja los puntos de conexión personalizados mediante la generación de un certificado en AWS Certificate Manager (ACM) o la importación de uno propio.

Puntos de conexión personalizados para nuevos dominios

Puede habilitar un punto de conexión personalizado para un nuevo dominio de OpenSearch Service mediante la consola de OpenSearch Service, la AWS CLI o la API de configuración.

Cómo personalizar el punto de conexión (consola)

1. En la consola de OpenSearch Service, seleccione Crear dominio y proporcione un nombre para el dominio.
2. En Punto de conexión personalizado, seleccione Habilitar punto de conexión personalizado.
3. En Nombre de anfitrión personalizado, ingrese el nombre de anfitrión del punto de conexión personalizado preferido. El nombre de anfitrión debe ser un nombre de dominio completo (FQDN), como www.yourdomain.com o example.yourdomain.com.

   > **Note**
   >
   > Si no dispone de un certificado comodín, es posible que necesite obtener un nuevo certificado para los subdominios del punto de conexión personalizado.

4. En Certificado de AWS, seleccione el certificado SSL que desee utilizar con el dominio. Si no hay certificados disponibles, puede importar uno a ACM o utilizar ACM para aprovisionar uno. Para más información, consulte Emisión y administración de certificados en la Guía del usuario de Certificate Manager de AWS.

   > **Note**
   >
   > El certificado debe tener el nombre del punto de conexión personalizado y debe estar en la misma cuenta que el dominio de OpenSearch Service. El estado del certificado debe ser EMITIDO.

   - Siga el resto de pasos para crear el dominio y elija Crear.
   - Seleccione el dominio, cuando se haya terminado de procesar, para ver el punto de conexión personalizado.

   Para utilizar la CLI o la API de configuración, utilice las operaciones `CreateDomain` y `UpdateDomainConfig`. Para más información, consulte AWS CLI Command Reference y Amazon OpenSearch Service API Reference.

Puntos de conexión personalizados para dominios existentes

Para agregar un punto de conexión personalizado a un dominio existente de OpenSearch Service, seleccione Editar y siga los pasos 2 a 4 anteriores.

Asignación CNAME

Después de habilitar un punto de conexión personalizado para el dominio de OpenSearch Service, puede crear una asignación CNAME en Amazon Route 53 (o en el proveedor de servicios DNS que prefiera). La creación de una asignación CNAME le permitirá dirigir el tráfico a su punto de conexión personalizado y sus subdominios. Sin esta asignación, no podrá dirigir el tráfico a su punto de conexión personalizado. A fin de conocer los pasos para crear esta asignación en Route 53, consulte Configuring DNS routing for a new domain y Creating a new hosted zone for a subdomain. Para otros proveedores, consulte su documentación.

Cree un registro CNAME en el que el punto de conexión personalizado apunte al punto de conexión del dominio generado automáticamente. Si el dominio es de doble pila, puede apuntar su registro CNAME a cualquiera de los dos puntos de conexión generados por el servicio. La capacidad de doble pila del punto de conexión personalizado depende del punto de conexión generado por el servicio al que apunte el registro CNAME. El nombre de host del punto de conexión personalizado es el nombre del registro CNAME y el nombre de host del punto de conexión del dominio es el valor del registro CNAME.

Si utiliza Autenticación SAML para OpenSearch Dashboards, debe actualizar su IdP con la nueva URL de SSO.

Puede utilizar Amazon Route 53 a fin de crear un tipo de registro de alias a fin de dirigir el punto de conexión personalizado de su dominio a un punto de conexión de búsqueda de doble pila. Para crear un tipo de registro de alias, debe configurar su dominio para que utilice el tipo de dirección IP de doble pila. Para ello, puede utilizar la API de Route 53.

Para crear un tipo de registro de alias mediante la API de Route 53, especifique el alias de destino de su dominio. Puede encontrar el alias de destino de su dominio en el campo Zona alojada (doble pila) de la sección de puntos de conexión personalizados de la consola de OpenSearch Service o al usar la API `DescribeDomain` y copiar el valor de `DomainEndpointV2HostedZoneId`.

Ajuste automático para Amazon OpenSearch Service

La función de ajuste automático en Amazon OpenSearch Service utiliza métricas de rendimiento y uso de un clúster de OpenSearch para sugerir cambios de configuración relacionados con la memoria, incluidos los tamaños de cola y caché y la configuración de máquina virtual Java (JVM) en los nodos. Estos cambios opcionales mejoran la velocidad y la estabilidad del clúster.

Algunos cambios se implementan inmediatamente, mientras que otros se programan durante la ventana de menor actividad de su dominio. Puede volver a la configuración predeterminada de OpenSearch Service en cualquier momento. A medida que la función de ajuste automático recopila y analiza las métricas de rendimiento del dominio, puede ver recomendaciones en la consola de OpenSearch Service en la página Notificaciones.

La función de ajuste automático está disponible en las Regiones de AWS comerciales para dominios que ejecutan cualquier versión de OpenSearch, o Elasticsearch 6.7 o posterior, con un tipo de instancia compatible.

Tipos de cambios

La función de ajuste automático cuenta con dos amplias categorías de cambios:

- Cambios no disruptivos que aplica a medida que se ejecuta el clúster.
- Cambios que requieren una implementación azul/verde, que se aplica durante el período de menor actividad del dominio.

Según las métricas de rendimiento del dominio, la función de ajuste automático puede sugerir realizar ajustes en las siguientes configuraciones:

Cambio de tipo	Categoría	Descripción
Tamaño de la pila de JVM	Blue/green (azul/verde)	De forma predeterminada, OpenSearch Service utiliza el 50 % de la RAM de una instancia para la pila de JVM, hasta un tamaño de 32 GiB. Aumentar este porcentaje proporciona más memoria a OpenSearch, pero deja menos memoria para el sistema operativo y otros procesos. Los valores más altos pueden disminuir el número

Cambio de tipo	Categoría	Descripción
		de pausas de recolección de basura, pero pueden aumentar la amplitud de esas pausas.
Configuración de JVM de nueva generación	Blue/green (azul/verde)	La configuración de JVM de “nueva generación” afecta la frecuencia de recolecciones de basura menores. Las recolecciones menores más frecuentes pueden disminuir el número de recolecciones principales y pausas.
Tamaño de la cola	No disruptivo	De forma predeterminada, el tamaño de la cola de búsqueda es `1000` y el tamaño de la cola de escritura es `10000`. La función de ajuste automático escala automáticamente las colas de búsqueda y escritura si hay una pila adicional disponible para encargarse de las solicitudes.
Tamaño de caché	No disruptivo	La caché de campo monitorea las estructuras de datos en la pila, por lo que es importante monitorear el uso de la caché. La función de ajuste automático escala el tamaño de la caché de datos de campo para evitar problemas de memoria y con los interruptores de circuito. La caché de solicitudes de partición se administra a nivel del nodo y tiene un tamaño máximo predeterminado del 1 % de la pila. La función de ajuste automático escala el tamaño de la caché de solicitudes de partición para aceptar más solicitudes de búsqueda e índice de las que puede manejar el clúster configurado.

Cambio de tipo	Categoría	Descripción
Solicitar tamaño	No disruptivo	De forma predeterminada, cuando el tamaño agregado de las solicitudes en vuelo supera el 10 % del total de JVM (2 % para tipos de instancia `t2` y 1 % para `t3.small`), OpenSearch llimita toda nueva solicitud `_search` y `_bulk` hasta que se completen las solicitudes existentes. Auto-Tune ajusta automáticamente este umbral, normalmente entre el 5-15 %, basándose en la cantidad de JVM que está actualmente ocupada en el sistema. Por ejemplo, si la presión de la memoria de la JVM es alta, el ajuste automático podría reducir el umbral al 5 %, momento en el que podría ver más rechazos hasta que el clúster se estabilice y el umbral aumente.

Habilitar o deshabilitar la función de ajuste automático

OpenSearch Service habilita la función de ajuste automático de forma predeterminada en nuevos dominios. Para habilitar o deshabilitar la función de ajuste automático en dominios existentes, recomendamos utilizar la consola, lo cual simplifica el proceso. Al habilitar la función de ajuste automático no se produce una implementación azul/verde.

Actualmente no se puede habilitar o desactivar la función de ajuste automático mediante AWS CloudFormation.

Consola

Cómo habilitar el ajuste automático en un dominio existente

1. Abra la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. En el panel de navegación, en Dominios, seleccione el nombre del dominio para abrir la configuración del clúster.
3. Seleccione Activar si el ajuste automático aún no está activado.
4. Si lo desea, seleccione Ventana de menor actividad para programar las optimizaciones que requieran una implementación azul/verde durante la ventana de menor actividad configurada en

el dominio. Para obtener más información, consulte the section called “Programación de mejoras de ajuste automático”.

5. Seleccione Guardar cambios.

CLI

Para habilitar el ajuste automático mediante el AWS CLI, envíe una solicitud UpdateDomainConfig:

```
aws opensearch update-domain-config \
  --domain-name my-domain \
  --auto-tune-options DesiredState=ENABLED
```

Programación de mejoras de ajuste automático

Antes del 16 de febrero de 2023, ajuste automático utilizaba ventanas de mantenimiento para programar los cambios que requerían una implementación azul/verde. Las ventanas de mantenimiento ahora están en desuso en favor del período de menor actividad, que es un bloque de tiempo diario de 10 horas durante el cual su dominio suele experimentar poco tráfico. Puede modificar la hora de inicio predeterminada para la ventana de menor actividad, pero no puede modificar la duración.

Todos los dominios que tenían habilitados los períodos de mantenimiento de ajuste automático antes de la introducción de las ventanas de menor actividad el 16 de febrero de 2023 pueden seguir utilizando las ventanas de mantenimiento antiguas sin interrupciones. Sin embargo, le recomendamos que migre sus dominios actuales para utilizar en su lugar la ventana de menor actividad para el mantenimiento de los dominios. Para obtener instrucciones, consulte the section called “Migración desde los intervalos de mantenimiento de ajuste automático”.

Consola

Cómo programar las acciones de ajuste automático en las ventanas de menor actividad

1. Abra la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. En el panel de navegación, en Dominios, seleccione el nombre del dominio para abrir la configuración del clúster.
3. Vaya a la pestaña Ajuste automático y seleccione Editar.
4. Seleccione Activar si el ajuste automático aún no está activado.

5. En Programar optimizaciones durante la ventana de menor actividad, seleccione Ventana de menor actividad.
6. Seleccione Guardar cambios.

CLI

Para configurar su dominio para programar acciones de ajuste automático durante la ventana de menor actividad configurada, incluya `UseOffPeakWindow` en la solicitud UpdateDomainConfig:

```
aws opensearch update-domain-config \
  --domain-name my-domain \
  --auto-tune-options
 DesiredState=ENABLED,UseOffPeakWindow=true,MaintenanceSchedules=null
```

Supervisión de cambios de ajuste automático

Puede supervisar las estadísticas de ajuste automático en Amazon CloudWatch. Para obtener una lista completa de las métricas, consulte the section called “Métricas de ajuste automático”.

OpenSearch Service envía eventos de ajuste automático a Amazon EventBridge. Puede utilizar EventBridge para configurar reglas que envíen un correo electrónico o realicen una acción específica cuando se reciba un evento. Para ver el formato de cada evento de ajuste automático enviado a EventBridge, consulte. the section called “Eventos de ajuste automático”

Etiquetado de dominios de Amazon OpenSearch Service

Las etiquetas permiten asignar información arbitraria a un dominio de Amazon OpenSearch Service para poder categorizar y filtrar esa información. Una etiqueta es un par de clave-valor que el usuario define y asocia a un dominio de OpenSearch Service. Estas etiquetas se pueden utilizar para hacer un seguimiento de los costos, agrupando los gastos correspondientes a recursos con etiqueta similar. AWS no aplica ningún significado semántico a las etiquetas. Las etiquetas se interpretan estrictamente como cadenas de caracteres. Todas las etiquetas presentan los siguientes elementos:

Elemento de etiqueta	Descripción	Obligatorio
Clave de etiqueta	La clave de la etiqueta es el nombre de la etiqueta. La clave debe ser exclusiva del dominio del servicio OpenSearch al que está asociada.	Sí

Elemento de etiqueta	Descripción	Obligatorio
	Para obtener una lista de restricciones básicas relativas a las claves y valores de las etiquetas, consulte Restricciones de las etiquetas definidas por el usuario.	
Valor de etiqueta	El valor de etiqueta es un valor de cadena en la etiqueta. Los valores de etiquetas pueden ser `null` y no tienen que ser únicos dentro de un conjunto de etiquetas. Por ejemplo, puede tener un par valor de clave en un conjunto de etiquetas en proyecto/Trinity y centro-de-costos/Trinity. Para obtener una lista de restricciones básicas relativas a las claves y valores de las etiquetas, consulte Restricciones de las etiquetas definidas por el usuario.	No

Cada dominio de OpenSearch Service de tiene un conjunto de etiquetas con todas las etiquetas asignadas a ese dominio de OpenSearch Service. AWS no asigna automáticamente ninguna etiqueta a los dominios de OpenSearch Service. Un conjunto de etiquetas puede contener entre 0 y 50 etiquetas. Si agrega una etiqueta a un dominio con la misma clave que una etiqueta existente en un recurso, el nuevo valor sobrescribirá el antiguo.

Ejemplos de etiquetas

Puede utilizar una clave para definir una categoría, y el valor podría ser un elemento dentro de esa categoría. Por ejemplo, puede definir una clave de etiqueta `project` y un valor de etiqueta `Salix` para indicar que el dominio de OpenSearch Service va asignado al proyecto Salix. También podría utilizar etiquetas para designar dominios de OpenSearch Service para pruebas o para producción mediante una clave como `environment=test` o `environment=production`. Intente utilizar un conjunto coherente de claves de etiquetas que facilite el seguimiento de los metadatos asociados a los dominios de OpenSearch Service.

También puede usar etiquetas para organizar la factura de AWS de modo que refleje su propia estructura de costos. Para ello, inscríbase para obtener una factura de Cuenta de AWS que incluya valores de clave de etiquetas. A continuación, para ver los costos de los recursos combinados, organice la información de facturación de acuerdo con los recursos que tienen los mismos valores de clave de etiqueta. Por ejemplo, puede etiquetar varios dominios de OpenSearch Service con pares clave-valor y, a continuación, organizar la información de facturación de modo que se muestre el

costo total de cada dominio para diversos servicios. Para obtener más información, consulte Uso de etiquetas de asignación de costos en la documentación de administración de costos y facturación de AWS.

> **Note**
>
> Las etiquetas se almacenan en caché con fines de autorización. Por este motivo, cuando se actualizan o se agregan valores a las etiquetas de los dominios de OpenSearch Service, pueden tardar varios minutos en estar disponibles.

Etiquetado de dominios (consola)

La consola es la forma más sencilla de etiquetas de un dominio.

Para crear una etiqueta (consola)

1. Visite https://aws.amazon.com y, a continuación, seleccione Iniciar sesión en la consola.
2. En Análisis, seleccione Amazon OpenSearch Service.
3. Seleccione el dominio al que desee agregar etiquetas y vaya a la pestaña Tags (Etiquetas).
4. Seleccione Administrar y Add new tag (Agregar nueva etiqueta).
5. Introduzca una clave de etiqueta y un valor opcional.
6. Seleccione guardar.

Para eliminar una etiqueta, siga los mismos pasos y elija Quitar en la página Administrar etiquetas.

Para obtener más información sobre cómo utilizar la consola para trabajar con etiquetas, consulte Tag Editor en la Guía de introducción a la consola de administración de AWS.

Etiquetado de dominios (AWS CLI)

Puede crear etiquetas de recurso usando la AWS CLI con el comando --add-tags.

Sintaxis

```
add-tags --arn=<domain_arn> --tag-list Key=<key>,Value=<value>
```

Parámetro	Descripción
`--arn`	Nombre de recurso de Amazon del dominio de OpenSearch Service al que se ha adjuntado la etiqueta.
`--tag-list`	Conjunto de pares de clave-valor separados por espacios con el siguiente formato: `Key=<key>,Value=<value>`

Ejemplo

En el siguiente ejemplo se crean dos etiquetas para el dominio registros:

```
aws opensearch add-tags --arn arn:aws:es:us-east-1:379931976431:domain/logs --tag-list
 Key=service,Value=OpenSearch Key=instances,Value=m3.2xlarge
```

Puede eliminar etiquetas de un dominio de OpenSearch Service ejecutando el comando --remove-tags.

Sintaxis

```
remove-tags --arn=<domain_arn> --tag-keys Key=<key>,Value=<value>
```

Parámetro	Descripción
`--arn`	Nombre de recurso de Amazon (ARN) del dominio de OpenSearch Service al que se ha adjuntado la etiqueta.
`--tag-keys`	Conjunto de pares clave-valor separados por espacios que se van a eliminar del dominio de OpenSearch Service.

Ejemplo

En el siguiente ejemplo se eliminan dos etiquetas del dominio logs que se crearon en el ejemplo anterior:

```
aws opensearch remove-tags --arn arn:aws:es:us-east-1:379931976431:domain/logs --tag-
keys service instances
```

Puede consultar las etiquetas existentes de un dominio de OpenSearch Service con el comando --list-tags:

Sintaxis

```
list-tags --arn=<domain_arn>
```

Parámetro	Descripción
`--arn`	Nombre de recurso de Amazon (ARN) del dominio de OpenSearch Service al que se han adjuntado las etiquetas.

Ejemplo

En el siguiente ejemplo se muestran todas las etiquetas de recursos del dominio logs:

```
aws opensearch list-tags --arn arn:aws:es:us-east-1:379931976431:domain/logs
```

Etiquetado de dominios (AWS SDK)

Los AWS SDK (excepto los SDK de Android e iOS) admiten todas las acciones definidas en la Referencia de la API de Amazon OpenSearch Service, incluidas las operaciones `AddTags`, `ListTags` y `RemoveTags`. Para obtener más información acerca de cómo instalar y utilizar los SDK de AWS, consulte los Kits de desarrollo de software de AWS.

Python

En este ejemplo, se utiliza el cliente OpenSearchService de bajo nivel de Python del AWS SDK para Python (Boto) para agregar una etiqueta a un dominio, enumerar la etiqueta adjunta al dominio y eliminar una etiqueta del dominio. Debe proporcionar valores para `DOMAIN_ARN`, `TAG_KEY` y `TAG_VALUE`.

```
import boto3
from botocore.config import Config  # import configuration

DOMAIN_ARN = ''  # ARN for the domain. i.e "arn:aws:es:us-east-1:123456789012:domain/
my-domain
TAG_KEY = ''  # The name of the tag key. i.e 'Smileyface'
TAG_VALUE = ''  # The value assigned to the tag. i.e 'Practicetag'
```

```
# defines the configurations parameters such as region

my_config = Config(region_name='us-east-1')
client = boto3.client('opensearch', config=my_config)


# defines the client variable

def addTags():
    """Adds tags to the domain"""

    response = client.add_tags(ARN=DOMAIN_ARN,
                               TagList=[{'Key': TAG_KEY,
                                         'Value': TAG_VALUE}])

    print(response)


def listTags():
    """List tags that have been added to the domain"""

    response = client.list_tags(ARN=DOMAIN_ARN)
    print(response)


def removeTags():
    """Remove tags that have been added to the domain"""

    response = client.remove_tags(ARN=DOMAIN_ARN, TagKeys=[TAG_KEY])

    print('Tag removed')
    return response
```

Ejecución de acciones administrativas en dominios de Amazon OpenSearch Service

Amazon OpenSearch Service ofrece varias opciones administrativas que proporcionan un control detallado si necesita solucionar problemas con su dominio. Estas opciones incluyen la posibilidad de reiniciar el proceso de OpenSearch en un nodo de datos y la posibilidad de reiniciar un nodo de datos.

OpenSearch Service supervisa los parámetros de estado de los nodos y, cuando hay anomalías, toma medidas correctivas para mantener la estabilidad de los dominios. Con las opciones administrativas para reiniciar el proceso de OpenSearch en un nodo y reiniciar el nodo mismo, puede controlar algunas de estas acciones de mitigación.

Puede usar el Consola de administración de AWS, la AWS CLI, o el AWS SDK para realizar estas acciones. En las siguientes secciones, se explica cómo realizar estas acciones con la consola.

Reiniciar el proceso de OpenSearch en un nodo en Amazon OpenSearch Service

Cómo reiniciar el proceso de OpenSearch en un nodo

1. Vaya a la consola de OpenSearch Service en https://console.aws.amazon.com/aos/.
2. En el panel de navegación izquierdo, seleccione Dominios. Seleccione el nombre del dominio con el que desea trabajar.
3. Cuando se abra la página de detalles del dominio, vaya a la pestaña Estado de la instancia.
4. En Nodos de datos, seleccione el botón situado junto al nodo en el que desea reiniciar el proceso.
5. Seleccione el menú desplegable Acciones y seleccione Reiniciar el proceso de OpenSearch/ElasticSearch.
6. Seleccione Confirmar en el modal.
7. Para ver el estado de la acción que ha iniciado, seleccione el nombre del nodo. Cuando se abra la página de detalles del nodo, seleccione la pestaña Eventos situada debajo del nombre del nodo para ver una lista de los eventos asociados a ese nodo.

Reiniciar un nodo de datos en Amazon OpenSearch Service

Cómo reiniciar un nodo de datos

1. Vaya a la consola de OpenSearch Service en https://console.aws.amazon.com/aos/.
2. En el panel de navegación izquierdo, seleccione Dominios. Seleccione el nombre del dominio con el que desea trabajar.
3. Cuando se abra la página de detalles del dominio, vaya a la pestaña Estado de la instancia.
4. En Nodos de datos, seleccione el botón situado junto al nodo en el que desea reiniciar el proceso.

5. Seleccione el menú desplegable Acciones y elija Nodo de reinicio.
6. Seleccione Confirmar en el modal.
7. Para ver el estado de la acción que ha iniciado, seleccione el nombre del nodo. Cuando se abra la página de detalles del nodo, seleccione la pestaña Eventos situada debajo del nombre del nodo para ver una lista de los eventos asociados a ese nodo.

Reiniciar el proceso de OpenSearch Dashboards en un nodo en Amazon OpenSearch Service

Puede reiniciar el proceso de OpenSearch Dashboards (anteriormente Kibana) para recuperarse de problemas como una interfaz bloqueada, errores de carga o visualizaciones que no responden. La opción de reiniciar los OpenSearch Dashboards solo está disponible para el nodo que ejecuta activamente el proceso de Dashboards. En la mayoría de los dominios de OpenSearch Service, este proceso se ejecuta en nodos coordinadores dedicados, no en nodos de datos. Como resultado, al abrir el menú desplegable Acciones en la consola, la opción normalmente aparece solo para los nodos coordinadores. Para obtener más información, consulte the section called “Nodos coordinadores dedicados”.

Este comportamiento depende de cómo esté configurado el dominio.

- Dominios con nodos coordinadores dedicados: el proceso de Dashboards se ejecuta exclusivamente en esos nodos y solo en ellos se muestra la opción de reinicio.
- Dominios sin nodos coordinadores dedicados: en configuraciones más simples o antiguas, es posible que los Dashboards se ejecuten en un nodo de datos y allí aparezca la opción de reinicio.
- Nodos maestros: estos nodos sirven únicamente para administrar el estado y las elecciones del clúster. No ejecutan Dashboards y nunca muestran la opción de reinicio.

Para determinar qué nodo ejecuta el proceso de Dashboards, vaya a la sección de configuración del clúster de su dominio y revise las funciones de los nodos. La opción de reiniciar solo está disponible para el nodo que aloja el proceso de Dashboards.

Para reiniciar el proceso de Dashboards en un nodo

1. Vaya a la consola de OpenSearch Service en https://console.aws.amazon.com/aos/.
2. En el panel de navegación izquierdo, seleccione Dominios. Seleccione el nombre del dominio con el que desea trabajar.

3. Cuando se abra la página de detalles del dominio, vaya a la pestaña Estado de la instancia.
4. En la sección para los nodos que están ejecutando el proceso de Dashboards, seleccione el botón al lado del nodo en el que usted desea reiniciar el proceso.
5. Seleccione el menú desplegable Acciones y elija Reiniciar el proceso Dashboard/Kibana.
6. Seleccione Confirmar en el modal.
7. Para ver el estado de la acción que ha iniciado, seleccione el nombre del nodo. Cuando se abra la página de detalles del nodo, seleccione la pestaña Eventos situada debajo del nombre del nodo para ver una lista de los eventos asociados a ese nodo.

Limitaciones

Las opciones administrativas presentan las siguientes limitaciones:

- Las opciones administrativas se admiten en las versiones 7.x y superiores de Elasticsearch.
- Las opciones administrativas no admiten dominios con Multi-AZ con modo de espera activado.
- El reinicio de los procesos de OpenSearch y Elasticsearch y el reinicio de los nodos de datos se admiten en dominios con tres o más nodos de datos.
- La compatibilidad con los procesos de Dashboards y Kibana se admite en dominios con dos o más nodos de datos.
- Para reiniciar el proceso de OpenSearch en un nodo o reiniciar un nodo, el dominio no debe estar en estado rojo y todos los índices deben tener las réplicas configuradas.

Uso de consultas directas de Amazon OpenSearch Service

Utilice la consulta directa de Amazon OpenSearch Service para analizar los datos de Registros de Amazon CloudWatch, Amazon S3 y Amazon Security Lake sin crear canalizaciones de ingesta. Esta integración sin ETL le permite consultar datos mediante OpenSearch SQL o PPL y explorarlos en Discover.

Para comenzar, configure el origen de datos en la consola de OpenSearch Service. Para Amazon S3, utilice conexiones de dominio y cree tablas con SQL en Query Workbench. Registros de CloudWatch y Security Lake utilizan orígenes y tablas de AWS Glue Data Catalog preconfiguradas.

Precios de las consultas directas

Cuando utilice las consultas directas de OpenSearch Service, incurrirá en cargos separados por OpenSearch Service y por los recursos utilizados para procesar y almacenar sus datos en Amazon S3, Registros de Amazon CloudWatch y Amazon Security Lake. A medida que realice consultas directas, verá los cargos por las unidades de cómputo de OpenSearch (OCU) por hora, que figuran en su factura según el tipo de uso de las OCU de DirectQuery.

Las consultas directas son de dos tipos: consultas de vista indexada e interactivas.

- Las consultas interactivas se utilizan para rellenar el selector de datos y realizar análisis de los datos en S3, Registros de CloudWatch o Security Lake.

  En el caso de las consultas directas de Amazon S3, cuando se ejecuta una nueva consulta desde Discover, OpenSearch Service inicia una nueva sesión que dura un mínimo de tres minutos. OpenSearch Service mantiene esta sesión activa para garantizar que las consultas posteriores se ejecuten rápidamente.

  En el caso de las consultas de Registros de CloudWatch y Security Lake, OpenSearch Service gestiona cada consulta con una tarea precalentada independiente, sin necesidad de mantener una sesión prolongada.
- Las consultas de vista indexada utilizan la computación para mantener las vistas indexadas en OpenSearch Service. Estas consultas suelen tardar más porque ingieren una cantidad de datos variable en un índice con nombre. Al indexar los datos, puede hacer que las futuras consultas interactivas se ejecuten más rápido o desbloquear capacidades de análisis avanzadas, como paneles o alertas, que requieren un índice como referencia.

En el caso de los orígenes de datos de Amazon S3, los datos indexados se almacenan en un dominio en función del tipo de instancia adquirido. Para los orígenes de datos conectados a Registros de CloudWatch y Security Lake, los datos indexados se almacenan en una colección de OpenSearch sin servidor en la que se cobra por los datos indexados (IndexingOCU), los datos buscados (SearchOCU) y los datos almacenados en GB.

Para obtener más información, consulte las secciones de Consulta directa y Sin servidor en Precios de Amazon OpenSearch Service.

Limitaciones de las consultas directas

Limitaciones generales

Las siguientes limitaciones se aplican a las consultas directas de OpenSearch Service.

- Algunos tipos de datos no son compatibles. Los tipos de datos admitidos se limitan a Parquet, CSV y JSON.
- Si la estructura de los datos cambia con el tiempo, tendrá que actualizar las vistas indexadas o las integraciones listas para usar a fin de tener en cuenta los cambios en la estructura de los datos.
- Las plantillas de AWS CloudFormation aún no son compatibles.
- Las instrucciones de OpenSearch SQL y OpenSearch PPL tienen limitaciones diferentes cuando se trabaja con índices de OpenSearch en comparación con el uso de consultas directas. La consulta directa admite comandos avanzados como JOIN, subconsultas y búsquedas, mientras que la compatibilidad con estos comandos en los índices de OpenSearch es limitada o inexistente. Para obtener más información, consulte the section called “Comandos SQL y PPL compatibles”.

Limitaciones de Amazon S3

Si consulta datos directamente en Amazon S3, se aplican las siguientes limitaciones adicionales:

- La consulta directa para S3 solo está disponible en los dominios de OpenSearch Service que ejecuten la versión 2.13 de OpenSearch o una posterior y requiere acceso a AWS Glue Data Catalog. Las tablas de AWS Glue Data Catalog existentes se deben volver a crear con SQL en OpenSearch Query Workbench.

- La consulta directa para S3 requiere que se especifique un bucket de puntos de control en Amazon S3. Este bucket mantiene el estado de las vistas indexadas, incluida la hora de la última actualización y los datos ingeridos más recientemente.
- Su dominio de OpenSearch y AWS Glue Data Catalog deben estar en la misma Cuenta de AWS. El bucket de S3 puede estar en una cuenta diferente (es necesario agregar una condición a la política de IAM), pero debe estar en la misma Región de AWS que el dominio.
- Las consultas directas de OpenSearch Service con S3 solo admiten tablas de Spark generadas desde Query Workbench. Spark Streaming no admite las tablas generadas con AWS Glue Data Catalog o Athena, que es necesario para mantener las vistas indexadas.
- Los tipos de instancias de OpenSearch tienen limitaciones de carga útil de red de 10 MiB o 100 MiB, según el tipo de instancia específico que elija.

Limitaciones de Registros de Amazon CloudWatch

Si consulta datos directamente en Registros de CloudWatch, se aplican las siguientes limitaciones adicionales:

- La integración de consultas directas con Registros de CloudWatch solo está disponible en las colecciones de OpenSearch Service y en la interfaz de usuario de OpenSearch.
- Las colecciones de OpenSearch sin servidor tienen limitaciones de carga útil de red de 100 MiB.
- Registros de CloudWatch admite las integraciones de flujo de VPC, CloudTrail y paneles de AWS WAF instaladas desde la consola.

Limitaciones de Amazon Security Lake

Si consulta datos directamente en Security Lake, se aplican las siguientes limitaciones adicionales:

- La integración de consultas directas con Security Lake solo está disponible en las colecciones de OpenSearch Service y en la interfaz de usuario de OpenSearch.
- Las colecciones de OpenSearch sin servidor tienen limitaciones de carga útil de red de 100 MiB.
- La administración de las tablas de Security Lake se realiza en Lake Formation.
- Security Lake solo admite vistas materializadas como vistas indexadas. No se admiten índices de cobertura.

Recomendaciones para el uso de consultas directas en Amazon OpenSearch Service

En esta página, se proporcionan recomendaciones para utilizar las consultas directas de Amazon OpenSearch Service con fines de análisis de datos de Registros de CloudWatch, Amazon S3 y Amazon Security Lake. Estas prácticas recomendadas lo ayudan a optimizar el rendimiento y a garantizar una consulta eficiente sin necesidad de ingerir ni duplicar datos.

Temas

- Recomendaciones generales
- Recomendaciones para Amazon S3
- Recomendaciones para Registros de CloudWatch
- Recomendaciones para Security Lake

Recomendaciones generales

Se recomienda que lleve a cabo las siguientes acciones al utilizar la consulta directa:

- Utilice la función `COALESCE SQL` para administrar las columnas que faltan y garantizar que se devuelvan los resultados.

- Utilice límites en sus consultas para asegurarse de no recuperar demasiados datos.
- Si planea analizar el mismo conjunto de datos muchas veces, cree una vista indexada para ingerir e indexar todos los datos en OpenSearch y elimínela cuando haya completado el análisis.
- Elimine los índices y las tareas de aceleración cuando ya no sean necesarios.
- No se admiten consultas que contengan nombres de campo idénticos pero que solo difieran en mayúsculas y minúsculas (como `field1` y `FIELD1`).

  Por ejemplo, no se admiten las siguientes consultas:

```
Select AWSAccountId, AwsAccountId from LogGroup
Select a.@LogStream, b.@logStream from Table A INNER Join Table B ona.id = b.id
```

  Sin embargo, se admite la siguiente consulta porque el nombre del campo (@logStream) es idéntico en ambos grupos de registros:

```
Select a.@logStream, b.@logStream from Table A INNER Join Table B on a.id = b.id
```

- Las funciones y expresiones deben funcionar con los nombres de los campos y formar parte de una sentencia `SELECT` con un grupo de registros especificado en la cláusula `FROM`.

  Por ejemplo, no se admite esta consulta:

  ```
  SELECT cos(10) FROM LogGroup
  ```

  Se admite esta consulta:

  ```
  SELECT cos(field1) FROM LogGroup
  ```

Recomendaciones para Amazon S3

Si utiliza Amazon OpenSearch Service para consultar datos directamente en Amazon S3, también le recomendamos lo siguiente:

- Ingiera datos en Amazon S3 mediante formatos de partición de año, mes, día y hora para acelerar las consultas.
- Al crear índices de omisión, utilice filtros Bloom para obtener campos con cardinalidad alta e índices con valores mínimos y máximos para rangos de valores amplios. Para los campos de cardinalidad alta, considere la posibilidad de utilizar un enfoque basado en valores para mejorar la eficiencia de las consultas.
- Utilice Administración de estados de índice para mantener el almacenamiento de las vistas materializadas y los índices de cobertura.

Recomendaciones para Registros de CloudWatch

Si utiliza Amazon OpenSearch Service para dirigir las consultas de datos en Registros de CloudWatch, también le recomendamos lo siguiente:

- Al buscar varios grupos de registros en una consulta, utilice la sintaxis adecuada. Para obtener más información, consulte the section called “Funciones de grupos de registros múltiples”.
- Cuando utilice comandos SQL o PPL, encierre determinados campos entre comillas invertidas para consultarlos correctamente. Las comillas invertidas son necesarias para los campos

con caracteres especiales (no alfabéticos ni numéricos). Por ejemplo, encierre `@message`, `Operation.Export`, y `Test::Field` entre comillas invertidas. No es necesario incluir las columnas con nombres exclusivamente alfabéticos entre comillas invertidas.

Ejemplo de consulta con campos sencillos:

```
SELECT SessionToken, Operation, StartTime  FROM `LogGroup-A`
LIMIT 1000;
```

Consulta similar con comillas invertidas agregadas:

```
SELECT `@SessionToken`, `@Operation`, `@StartTime` FROM `LogGroup-A`
LIMIT 1000;
```

Recomendaciones para Security Lake

Si utiliza Amazon OpenSearch Service para realizar consultas directas de datos en Security Lake, también le recomendamos lo siguiente:

- Compruebe el estado de Security Lake y asegúrese de que funcione sin problemas. Para ver los pasos detallados de solución de problemas, consulte Solución de problemas del estado del lago de datos en la Guía del usuario de Amazon Security Lake.
- Compruebe el acceso a su consulta:
  - Si realiza consultas en Security Lake desde una cuenta diferente a la cuenta de administrador delegado de Security Lake, configure un suscriptor con acceso de consulta en Security Lake.
  - Si realiza consultas en Security Lake desde la misma cuenta, compruebe si hay algún mensaje en Security Lake acerca del registro de sus buckets de S3 administrados en LakeFormation.
- Explore las plantillas de consulta y los paneles prediseñados para iniciar su análisis con rapidez.
- Familiarícese con Open Cybersecurity Schema Framework (OCSF) y Security Lake:
  - Revise los ejemplos de asignación de esquemas para los orígenes de AWS en el repositorio de GitHub de OCSF.
  - A fin de aprender a realizar consultas de Security Lake de forma eficaz, visite las consultas de Security Lake para la versión 2 del origen de AWS (OCSF 1.1.0).
  - Mejore el rendimiento de las consultas mediante el uso de particiones: `accountid`, `region` y `time_dt`.

- Familiarícese con la sintaxis de SQL, que Security Lake admite para realizar consultas. Para obtener más información, consulte the section called “Comandos SQL compatibles”.

Cuotas de las consultas directas

Su cuenta incluye las siguientes cuotas en relación con las consultas directas de OpenSearch Service.

Cuotas para Amazon S3

Cada vez que inicie una consulta a un origen de datos de Amazon S3, OpenSearch Service abrirá una sesión y la mantendrá activa durante al menos tres minutos. Esto reduce la latencia de las consultas al eliminar el tiempo de inicio de la sesión en las consultas posteriores.

Descripción	Máximo	Puede anular
Conexiones por dominio	10	Sí
Origen de datos por dominio	20	Sí
Índices por dominio	5	Sí
Sesiones simultáneas por origen de datos	10	Sí
OCU máxima por consulta	60	Sí
Tiempo máximo de ejecución de consulta (minutos)	30	Sí
OCU máximas por aceleración	20	Sí
Almacenamiento efímero máximo	20	Sí

Cuotas para Registros de CloudWatch

> **Note**
>
> Si desea realizar consultas directas con Información de registros de CloudWatch, asegúrese de consultar the section called “Información de registros de CloudWatch”.

Descripción	Valor	¿El límite es flexible?	Notas
Límite de TPS de nivel de cuenta en las API de consultas directas	3 TPS	Sí	
Número máximo de orígenes de datos	20	Sí	El límite es por Cuenta de AWS.
Número máximo de índices o vistas materializadas que se actualizan automátic amente	30	Sí	El límite es por origen de datos.
Máximo de consultas simultáneas	15	Sí	El límite se aplica a las consultas en estado pendiente o en ejecución. Incluye consultas interactivas (por ejemplo, comandos de recuperación de datos como `SELECT`) y consultas de índice (por ejemplo, operaciones como `CREATE`, `ALTER` y `DROP`).
OCU simultáneas máximas por consulta	512	Sí	Unidades de cómputo de OpenSearch (OCU). El límite se basa en 15 ejecutores y 1 controlad or, cada uno con 16 vCPU y 32 GB de memoria.

Descripción	Valor	¿El límite es flexible?	Notas
			Representa la potencia de procesamiento simultáneo.
Tiempo máximo de ejecución de consulta en minutos	60	No	Se aplica un límite a las consultas de OpenSearch PPL o SQL en Información de registros de CloudWatch.
Periodo para eliminar los ID de consulta obsoletos	90 días	Sí	Este es el periodo de tiempo después del cual OpenSearch Service elimina los metadatos de las consultas de las entradas más antiguas. Por ejemplo, se produce un error al llamar a GetDirectQuery o GetDirectQueryResult en el caso de consultas de más de 90 días.

Cuotas para Security Lake

Descripción	Valor	¿El límite es flexible?	Notas
Límite de TPS de nivel de cuenta en las API de consultas directas	3 TPS	Sí	
Número máximo de orígenes de datos	20	Sí	El límite es por Cuenta de AWS.
Número máximo de índices o vistas materializadas que se actualizan automáticamente	30	Sí	Se aplica un límite por origen de datos. Solo incluye índices y vistas materializadas (MV) con la actualización automática establecida en “true”.

Descripción	Valor	¿El límite es flexible?	Notas
Máximo de consultas simultáneas	30	Sí	El límite se aplica a las consultas en estado pendiente o en ejecución. Incluye consultas interactivas (por ejemplo, comandos de recuperación de datos como `SELECT`) y consultas de índice (por ejemplo, operaciones como `CREATE`, `ALTER` y `DROP`).
OCU simultáneas máximas por consulta	512	Sí	Unidades de cómputo de OpenSearch (OCU). El límite se basa en 15 ejecutores y 1 controlador, cada uno con 16 vCPU y 32 GB de memoria. Representa la potencia de procesamiento simultáneo.
Tiempo máximo de ejecución de consulta en minutos	30	No	Se aplica solo a consultas interactivas (por ejemplo, comandos de recuperación de datos como `SELECT`). En el caso de las consultas `REFRESH`, el límite es de 6 horas.
Periodo para eliminar los ID de consulta obsoletos	90 días	Sí	Este es el periodo de tiempo después del cual OpenSearch Service elimina los metadatos de las consultas de las entradas más antiguas. Por ejemplo, se produce un error al llamar a GetDirectQuery o GetDirectQueryResult en el caso de consultas de más de 90 días.

Regiones de AWS compatible

Las consultas directas de OpenSearch Service en Amazon S3, Registros de CloudWatch y Security Lake se admiten en las siguientes Regiones de AWS:

Regiones de AWS disponibles para Amazon S3

- Asia-Pacífico (Hong Kong)
- Asia-Pacífico (Mumbai)
- Asia-Pacífico (Seúl)
- Asia-Pacífico (Singapur)
- Asia-Pacífico (Sídney)
- Asia-Pacífico (Tokio)
- Canadá (centro)
- Europa (Fráncfort)
- Europa (Irlanda)
- Europa (Estocolmo)
- Este de EE. UU. (Norte de Virginia)
- Este de EE. UU. (Ohio)
- Oeste de EE. UU. (Oregón)

Regiones de AWS disponibles para Registros de CloudWatch

- Asia-Pacífico (Mumbai)
- Asia-Pacífico (Hong Kong)
- Asia-Pacífico (Singapur)
- Asia-Pacífico (Sídney)
- Asia-Pacífico (Tokio)
- Canadá (centro)
- Europa (Fráncfort)
- Europa (Irlanda)
- Europa (Estocolmo)
- Este de EE. UU. (Norte de Virginia)
- Este de EE. UU. (Ohio)
- Oeste de EE. UU. (Oregón)
- Europa (París)

- Europa (Londres)
- América del Sur (São Paulo)

Regiones de AWS disponibles para Security Lake

- Asia-Pacífico (Mumbai)
- Asia-Pacífico (Singapur)
- Asia-Pacífico (Sídney)
- Asia-Pacífico (Tokio)
- Canadá (centro)
- Europa (Fráncfort)
- Europa (Irlanda)
- Europa (Estocolmo)
- Este de EE. UU. (Norte de Virginia)
- Este de EE. UU. (Ohio)
- Oeste de EE. UU. (Oregón)
- Europa (París)
- Europa (Londres)
- América del Sur (São Paulo)

Consulta directa de datos de Amazon S3 en OpenSearch Service

En esta sección, se explica el proceso de creación y configuración de una integración de origen de datos en Amazon OpenSearch Service, lo que le permitirá consultar y analizar de forma eficiente los datos almacenados en Amazon S3.

En las siguientes páginas, aprenderá a configurar un origen de datos de consulta directa de Amazon S3, conocerá los requisitos previos necesarios y podrá seguir los procedimientos paso a paso con la Consola de administración de AWS y la API de OpenSearch Service. También se describen los próximos pasos importantes, como la asignación de roles de AWS Glue Data Catalog y la configuración de los controles de acceso en OpenSearch Dashboards.

Temas

- Creación de una integración de origen de datos de Amazon S3 en OpenSearch Service
- Configuración y consulta de un origen de datos de S3 en OpenSearch Dashboards

Creación de una integración de origen de datos de Amazon S3 en OpenSearch Service

Puede crear un nuevo origen de datos de consulta directa de Amazon S3 para OpenSearch Service a través de la Consola de administración de AWS o la API. Cada nuevo origen de datos utiliza el AWS Glue Data Catalog para administrar las tablas que representan los buckets de Amazon S3.

Temas

- Requisitos previos
- Procedimiento
- Pasos a seguir a continuación
- Asignar el rol de AWS Glue Data Catalog
- Recursos adicionales

Requisitos previos

Antes de comenzar, asegúrese de que ha revisado la siguiente documentación:

- the section called “Limitaciones de Amazon S3”
- the section called “Recomendaciones para Amazon S3”
- the section called “Cuotas para Amazon S3”

Antes de poder crear un origen de datos, debe tener los siguientes recursos en su Cuenta de AWS:

- Un dominio de OpenSearch con versión 2.13 o una posterior. Esta es la base para configurar la integración de consultas directas. Para obtener instrucciones acerca de cómo configurarlo, consulte the section called “ Creación de dominios de OpenSearch Service”.
- Uno o varios buckets de S3. Deberá especificar los buckets que contienen los datos que desea consultar y un bucket en el cual almacenar los puntos de control de las consultas. Para obtener instrucciones acerca de cómo crear un bucket de S3, consulte Crear un bucket en la Guía del usuario de Amazon S3.

- (Opcional) Una o más tablas de AWS Glue. La consulta de datos en Amazon S3 requiere que tenga tablas configuradas en AWS Glue Data Catalog para que apunten a los datos de S3. Debe crear las tablas con OpenSearch Query Workbench. Las tablas de Hive existentes no son compatibles.

  Si es la primera vez que configura un origen de datos de Amazon S3, debe crear un origen de datos de administrador para configurar todas las tablas de AWS Glue Data Catalog. Para ello, instale las integraciones predefinidas de OpenSearch o utilice OpenSearch Query Workbench a fin de crear tablas SQL personalizadas para casos de uso avanzados. Si desea ver ejemplos acerca de cómo crear tablas para registros de VPC, CloudTrail y AWS WAF, consulte la documentación en GitHub para VPC, CloudTrail y AWS WAF. Tras crear las tablas, puede crear nuevos orígenes de datos de Amazon S3 y restringir el acceso a tablas limitadas.
- (Opcional) Un rol de IAM creado manualmente. Puede usar este rol para administrar el acceso al origen de datos. Como alternativa, puede hacer que OpenSearch Service le cree un rol automáticamente con los permisos necesarios. Si decide utilizar un rol de IAM creado manualmente, siga las instrucciones que se indican en the section called “Permisos necesarios para los roles de IAM creados manualmente”.

Procedimiento

Puede configurar un origen de datos de consulta directa en un dominio con la Consola de administración de AWS o la API de OpenSearch Service.

Para configurar un origen de datos mediante la Consola de administración de AWS

1. Vaya a la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/.
2. En el panel de navegación izquierdo, seleccione Dominios.
3. Seleccione el dominio para el que desea configurar un nuevo origen de datos. Se abrirá la página de detalles del dominio.
4. Seleccione la pestaña Conexiones que aparece debajo de los detalles generales del dominio y busque la sección Consulta directa.
5. Seleccione Configurar origen de datos.
6. Escriba un nombre y una descripción opcional para el nuevo origen de datos.
7. Seleccione Amazon S3 con AWS Glue Data Catalog.
8. En Configuración del permiso de acceso de IAM, elija cómo administrar el acceso.

a. Si desea crear automáticamente un rol para este origen de datos, siga estos pasos:

   i. Seleccione Crear un nuevo rol.
   ii. Ingrese un nombre para el rol de IAM.
   iii. Seleccione uno o más buckets de S3 que contengan datos que desee consultar.
   iv. Seleccione un bucket de S3 de puntos de control en el cual almacenar los puntos de control de las consultas.
   v. Seleccione una o más bases de datos o tablas de AWS Glue para definir qué datos se pueden consultar. Si aún no se han creado las tablas, proporcione acceso a la base de datos predeterminada.

b. Si desea utilizar un rol existente que usted administra, siga estos pasos:

   i. Seleccione Usar un rol existente.
   ii. Seleccione un rol existente en el menú desplegable.

> **Note**
>
> Cuando utilice su propio rol, debe asegurarse de que tiene todos los permisos requeridos. Para ello, adjunte las políticas necesarias desde la consola de IAM. Para obtener más información, consulte el ejemplo de política en the section called “Permisos necesarios para los roles de IAM creados manualmente”.

9. Elija Configurar. Esto abre la pantalla de detalles del origen de datos con una URL de OpenSearch Dashboards. Puede navegar hasta esta URL para completar los siguientes pasos.

API de OpenSearch Service

Utilice la operación de la API AddDataSource para crear un nuevo origen de datos en su dominio.

```
POST https://es.region.amazonaws.com/2021-01-01/opensearch/domain/domain-name/
dataSource

{
   "DataSourceType": {
        "S3GlueDataCatalog": {
            "RoleArn": "arn:aws:iam::account-id:role/role-name"
        }
```

```
    }
   "Description": "data-source-description",
   "Name": "my-data-source"
}
```

Pasos a seguir a continuación

Visitar OpenSearch Dashboards

Después de crear un origen de datos, OpenSearch Service le proporciona un enlace a OpenSearch Dashboards. Puede utilizar esto para configurar el control de acceso, definir tablas, instalar integraciones listas para usar y consultar sus datos.

Para obtener más información, consulte the section called “Configuración de un origen de datos de S3”.

Asignar el rol de AWS Glue Data Catalog

Si ha habilitado el control de acceso detallado después de crear un origen de datos, debe asignar los usuarios que no sean administradores a un rol de IAM con acceso a AWS Glue Data Catalog para poder ejecutar consultas directas. Para crear manualmente un rol `glue_access` de backend que pueda asignar al rol de IAM, siga estos pasos:

> **Note**
>
> Los índices se utilizan para cualquier consulta realizada en el origen de datos. Un usuario con acceso de lectura al índice de solicitudes de un origen de datos determinado puede leer todas las consultas realizadas en ese origen de datos. Un usuario con acceso de lectura al índice de resultados puede leer los resultados de todas las consultas realizadas en ese origen de datos.

1. Permanezca en el menú principal de OpenSearch Dashboards y elija Seguridad, Roles, Crear roles.
2. Asigne el rol glue_access.
3. En Permisos de clúster, seleccione `indices:data/write/bulk*`, `indices:data/read/scroll`, `indices:data/read/scroll/clear`.
4. En Índice, introduzca los siguientes índices a los que quiere conceder acceso al usuario con el rol:

- `.query_execution_request_<name of data source>`
- `query_execution_result_<name of data source>`
- `.async-query-scheduler`
- `flint_*`

5. En Permisos de índice, seleccione `indices_all`.
6. Elija Crear.
7. Seleccione Usuarios asignados, Administrar mapeo.
8. En Roles de backend, agregue el ARN del rol de AWS Glue que necesita permiso para llamar a su dominio.

```
arn:aws:iam::account-id:role/role-name
```

9. Seleccione Asignar y confirme que el rol aparece en Usuarios asignados.

Para obtener más información sobre la asignación de roles, consulte the section called “Mapear roles a usuarios”.

Recursos adicionales

Permisos necesarios para los roles de IAM creados manualmente

Al crear un origen de datos para su dominio, elige un rol de IAM para administrar el acceso a sus datos. Dispone de dos opciones para hacerlo:

1. Crear un nuevo rol de IAM automáticamente
2. Utilizar un rol de IAM creado manualmente

Si utiliza un rol creado manualmente, debe asociar los permisos correctos al rol. Los permisos del rol deben permitir el acceso al origen de datos específico y permitir que OpenSearch Service asuma el rol. Esto es necesario para que OpenSearch Service pueda acceder de forma segura a los datos e interactuar con ellos.

El siguiente ejemplo de política muestra los permisos con privilegios mínimos necesarios para crear y administrar un origen de datos. Si tiene permisos más generales, como `s3:*` o la política de `AdminstratorAccess`, estos engloban los permisos con privilegios mínimos de la política de ejemplo.

En la siguiente política de ejemplo, reemplace el *texto de marcador* con su información, según corresponda.

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Sid":"HttpActionsForOpenSearchDomain",
         "Effect":"Allow",
         "Action":"es:ESHttp*",
"Resource":"arn:aws:es:us-east-1:111122223333:domain/example.com/*"
      },
      {
         "Sid":"AmazonOpenSearchS3GlueDirectQueryReadAllS3Buckets",
         "Effect":"Allow",
         "Action":[
            "s3:GetObject",
            "s3:GetObjectVersion",
            "s3:ListBucket"
         ],
         "Condition":{
            "StringEquals":{
               "aws:ResourceAccount":"111122223333"
            }
         },
         "Resource":"*"
      },
      {
         "Sid":"AmazonOpenSearchDirectQueryGlueCreateAccess",
         "Effect":"Allow",
         "Action":[
            "glue:CreateDatabase",
            "glue:CreatePartition",
            "glue:CreateTable",
            "glue:BatchCreatePartition"
         ],
         "Resource":"*"
      },
      {
         "Sid":"AmazonOpenSearchS3GlueDirectQueryModifyAllGlueResources",
         "Effect":"Allow",
```

```
            "Action":[
               "glue:DeleteDatabase",
               "glue:DeletePartition",
               "glue:DeleteTable",
               "glue:GetDatabase",
               "glue:GetDatabases",
               "glue:GetPartition",
               "glue:GetPartitions",
               "glue:GetTable",
               "glue:GetTableVersions",
               "glue:GetTables",
               "glue:UpdateDatabase",
               "glue:UpdatePartition",
               "glue:UpdateTable",
               "glue:BatchGetPartition",
               "glue:BatchDeletePartition",
               "glue:BatchDeleteTable"
            ],
            "Resource":[
               "arn:aws:glue:us-east-1:111122223333:table/*",
               "arn:aws:glue:us-east-1:111122223333:database/*",
               "arn:aws:glue:us-east-1:111122223333:catalog",
               "arn:aws:es:us-east-1:111122223333:domain/domain_name"
            ],
            "Condition":{
               "StringEquals":{
                  "aws:ResourceAccount":"111122223333"
               }
            }
         },
         {
            "Sid":"ReadAndWriteActionsForS3CheckpointBucket",
            "Effect":"Allow",
            "Action":[
               "s3:ListMultipartUploadParts",
               "s3:DeleteObject",
               "s3:GetObject",
               "s3:PutObject",
               "s3:GetBucketLocation",
               "s3:ListBucket"
            ],
            "Condition":{
               "StringEquals":{
                  "aws:ResourceAccount":"111122223333"
```

```
            }
         },
         "Resource":[
            "arn:aws:s3:::amzn-s3-demo-bucket",
            "arn:aws:s3:::amzn-s3-demo-bucket/*"
         ]
      }
   ]
}
```

Para admitir buckets de Amazon S3 en diferentes cuentas, tendrá que incluir una condición en la política de Amazon S3 y agregar la cuenta adecuada.

En la siguiente condición de ejemplo, reemplace el *texto de marcador* con su información, según corresponda.

```
"Condition": {
               "StringEquals": {
                   "aws:ResourceAccount": "{{accountId}}"
               }
```

El rol debe tener la siguiente política de confianza, que especifica el ID de destino.

JSON

```
{
    "Version":"2012-10-17",
    "Statement":[
       {
          "Effect":"Allow",
          "Principal":{
             "Service": "directquery.opensearchservice.amazonaws.com"
          },
          "Action":"sts:AssumeRole"
       }
     ]
}
```

Para obtener instrucciones sobre cómo crear los roles, consulte Creación de un rol mediante políticas de confianza personalizadas.

Si tiene habilitado el control de acceso detallado en OpenSearch Service, se creará automáticamente un nuevo rol de control de acceso detallado de OpenSearch para su origen de datos. El nombre del nuevo rol de control de acceso detallado será `AWSOpenSearchDirectQuery` *`<name of data source>`*.

De forma predeterminada, el rol solo tiene acceso a los índices de orígenes de datos de consulta directa. Si bien puede configurar el rol para limitar o conceder el acceso a su origen de datos, se recomienda no ajustar el acceso de este rol. Si elimina el origen de datos, se eliminará este rol. Esto eliminará el acceso de los demás usuarios si están asignados al rol.

Configuración y consulta de un origen de datos de S3 en OpenSearch Dashboards

Ahora que ha creado su origen de datos, puede configurar los ajustes de seguridad, definir las tablas de Amazon S3 o configurar la indexación acelerada de datos. En esta sección, se explican varios casos de uso del origen de datos en OpenSearch Dashboards antes de realizar consultas en los datos.

Para configurar las siguientes secciones, primero debe navegar hasta el origen de datos en OpenSearch Dashboards. En el menú de navegación izquierdo, en Administración, elija Origen de datos. En Administrar orígenes de datos, seleccione el nombre del origen de datos que creó en la consola.

Creación de tablas de Spark con Query Workbench

Para las consultas directas de OpenSearch Service a Amazon S3 se utilizan las tablas de Spark incluidas en el AWS Glue Data Catalog. Puede crear tablas desde Query Workbench sin tener que salir de OpenSearch Dashboards.

Para administrar las bases de datos y tablas existentes en el origen de datos, o para crear tablas nuevas en las que desee utilizar consultas directas, seleccione Query Workbench en el menú de navegación de la izquierda y elija el origen de datos de Amazon S3 en el menú desplegable de orígenes de datos.

Para configurar una tabla para los registros de flujo de VPC almacenados en S3 en formato Parquet, ejecute la siguiente consulta:

```
CREATE TABLE
datasourcename.gluedatabasename.vpclogstable (version INT, account_id STRING,
 interface_id STRING,
srcaddr STRING, dstaddr STRING, srcport INT, dstport INT, protocol INT, packets
 BIGINT,
bytes BIGINT, start BIGINT, end BIGINT, action STRING, log_status STRING,
`aws-account-id` STRING, `aws-service` STRING, `aws-region` STRING, year STRING,
month STRING, day STRING, hour STRING)

USING parquet PARTITIONED BY (aws-account-id, aws-service, aws-region, year, month,
day, hour)

LOCATION "s3://accountnum-vpcflow/AWSLogs"
```

Tras crear la tabla, ejecute la siguiente consulta para asegurarse de que es compatible con las consultas directas:

```
MSCK REPAIR TABLE  datasourcename.databasename.vpclogstable
```

Configuración de las integraciones para tipos de registro de AWS populares

Puede integrar los tipos de registro de AWS almacenados en Amazon S3 con OpenSearch Service. Utilice OpenSearch Dashboards para instalar integraciones que crean tablas de AWS Glue Data Catalog, consultas guardadas y paneles. Estas integraciones utilizan vistas indexadas para mantener los paneles actualizados.

Para obtener instrucciones acerca de cómo instalar una integración, consulte Instalación de un activo de integración en la documentación de OpenSearch.

Cuando seleccione una integración, asegúrese de que tenga la etiqueta `S3 Glue`.

Al configurar la integración, especifique Conexión a S3 para el tipo de conexión. A continuación, seleccione el origen de datos para la integración, la ubicación de Amazon S3 de los datos, el punto de control para administrar la indexación de la aceleración y los activos necesarios para su caso de uso.

> **Note**
>
> Asegúrese de que el bucket de S3 de su punto de control tenga permisos de escritura para la ubicación del punto de control. Sin estos permisos, se producirá un error en las aceleraciones de la integración.

Cómo configurar el control de acceso

En la página de detalles de su origen de datos, busque la sección Controles de acceso y elija Editar. Si el dominio tiene habilitado un control de acceso detallado, elija Restringido y seleccione las funciones a las que desee proporcionar acceso al nuevo origen de datos. También puede elegir Solo administrador si solo quiere que el administrador tenga acceso al origen de datos.

> **Important**
>
> Los índices se utilizan para cualquier consulta realizada en el origen de datos. Un usuario con acceso de lectura al índice de solicitudes de un origen de datos determinado puede leer todas las consultas realizadas en ese origen de datos. Un usuario con acceso de lectura al índice de resultados puede leer los resultados de todas las consultas realizadas en ese origen de datos.

Consulta de datos de S3 en OpenSearch Discover

Después de configurar las tablas y configurar la aceleración de consultas opcional deseada, ya puede empezar a analizar sus datos. Para consultar sus datos, seleccione el origen de datos en el menú desplegable. Si utiliza Amazon S3 y OpenSearch Dashboards, vaya a Discover y seleccione el nombre del origen de datos.

Si utiliza un índice de omisión o no ha creado uno, puede utilizar SQL o PPL para consultar los datos. Si ha configurado una vista materializada o un índice de cobertura, ya tiene un índice y puede usar el lenguaje de consultas de Dashboards (DQL) en todo Dashboards. También puede usar PPL con el complemento Observabilidad y SQL con el complemento Query Workbench. Actualmente, solo los complementos Observabilidad y Query Workbench admiten PPL y SQL. Para consultar datos mediante la API de OpenSearch Service, consulte la documentación de la API asíncrona.

> **Note**
>
> No se admiten todas las instrucciones, comandos y funciones de SQL y PPL. Para obtener una lista de los comandos admitidos, consulte the section called “Comandos SQL y PPL compatibles”.
> Si ha creado una vista materializada o un índice de cobertura, puede utilizar DQL para consultar sus datos, siempre que los haya indexado dentro.

Solución de problemas

Puede haber casos en los que los resultados no devuelvan los resultados esperados. Si tiene algún problema, asegúrese de seguir las the section called “Recomendaciones”.

Consulta directa de datos de Registros de Amazon CloudWatch en OpenSearch Service

En esta sección, se explica el proceso de creación y configuración de una integración de origen de datos en Amazon OpenSearch Service, lo que le permitirá consultar y analizar de forma eficiente los datos almacenados en Registros de CloudWatch.

En las siguientes páginas, aprenderá a configurar un origen de datos de consulta directa de Registros de CloudWatch, conocerá los requisitos previos necesarios y podrá seguir los procedimientos paso a paso con la Consola de administración de AWS.

Temas

- Creación de una integración de origen de datos de Registros de Amazon CloudWatch en OpenSearch Service
- Configuración y consulta de un origen de datos de Registros de CloudWatch en OpenSearch Dashboards

Creación de una integración de origen de datos de Registros de Amazon CloudWatch en OpenSearch Service

Si utiliza Amazon OpenSearch sin servidor para sus necesidades de observabilidad, ahora puede analizar sus Registros de Amazon CloudWatch sin copiar ni ingerir los datos en OpenSearch Service.

Esta capacidad aprovecha la consulta directa para consultar datos, de forma similar al análisis de datos en Amazon S3 desde OpenSearch Service. Para comenzar, puede crear un nuevo origen de datos conectado desde la Consola de administración de AWS.

Puede crear un nuevo origen de datos para analizar los datos de Registros de CloudWatch sin tener que configurar Amazon OpenSearch sin servidor para consultar directamente los registros operativos en Registros de CloudWatch. Esto le permite analizar los datos operativos a los que ha accedido y que se encuentran fuera de OpenSearch Service. Al realizar consultas en OpenSearch Service y Registros de CloudWatch, puede empezar a analizar los registros en Registros de CloudWatch y, a continuación, volver a supervisar los orígenes de datos en OpenSearch sin tener que cambiar de herramienta.

Para utilizar esta característica, cree un origen de datos de consulta directa de Registros de CloudWatch para OpenSearch Service a través de la Consola de administración de AWS.

Temas

- Requisitos previos
- Procedimiento
- Pasos a seguir a continuación
- Recursos adicionales

Requisitos previos

Antes de comenzar, asegúrese de que ha revisado la siguiente documentación:

- the section called “Limitaciones de Registros de Amazon CloudWatch”
- the section called “Recomendaciones para Registros de CloudWatch”
- the section called “Cuotas para Registros de CloudWatch”

Antes de poder crear un origen de datos, debe tener los siguientes recursos en su Cuenta de AWS:

- Habilite Registros de CloudWatch. Configure Registros de CloudWatch para recopilar registros en la misma Cuenta de AWS que su recurso de OpenSearch. Para obtener más información, consulte Introducción a Registros de CloudWatch en la guía del usuario de Registros de Amazon CloudWatch.
- Uno o más grupos de registros de CloudWatch.Puede especificar los grupos de registros que contienen los datos que desea consultar. Para obtener información acerca de cómo crear un

grupo de registros, consulte Crear un grupo de registro en Registros de CloudWatch en la guía del usuario de Registros de Amazon CloudWatch.

- (Opcional) Un rol de IAM creado manualmente.Puede usar este rol para administrar el acceso al origen de datos. Como alternativa, puede hacer que OpenSearch Service le cree un rol automáticamente con los permisos necesarios. Si decide utilizar un rol de IAM creado manualmente, siga las instrucciones que se indican en the section called “Permisos necesarios para los roles de IAM creados manualmente”.

Procedimiento

Puede configurar un origen de datos de consulta de nivel de colección con la Consola de administración de AWS.

Para configurar un origen de datos de nivel de colección mediante la Consola de administración de AWS

1. Vaya a la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/.
2. En el panel de navegación de la izquierda, vaya a Administración central y seleccione Orígenes de datos conectados.
3. Elija Conectar.
4. Elija CloudWatch como tipo de origen de datos.
5. Elija Siguiente.
6. En Detalles de la conexión de datos, escriba un nombre y una descripción opcional.
7. En Rol de IAM, elija cómo administrar el acceso a los grupos de registros.
    a. Si desea crear automáticamente un rol para este origen de datos, siga estos pasos:
        i. Seleccione Crear un nuevo rol.
        ii. Ingrese un nombre para el rol de IAM.
        iii. Seleccione uno o más grupos de registros para definir qué datos se pueden consultar.
    b. Si desea utilizar un rol existente que usted administra, siga estos pasos:
        i. Seleccione Usar un rol existente.
        ii. Seleccione un rol existente en el menú desplegable.

> **Note**
>
> Cuando utilice su propio rol, debe asegurarse de que tiene todos los permisos requeridos. Para ello, adjunte las políticas necesarias desde la consola de IAM. Para obtener más información, consulte the section called “Permisos necesarios para los roles de IAM creados manualmente”.

8. (Opcional) En Etiquetas, agregue etiquetas al origen de datos.
9. Elija Siguiente.
10. En Configurar OpenSearch, seleccione cómo configurar OpenSearch.
    a. Utilice la configuración predeterminada:
       - Revise los nombres de los recursos y la configuración de retención de datos predeterminados. Le sugerimos que utilice nombres personalizados.

         Al utilizar la configuración predeterminada, se crea una nueva aplicación OpenSearch y un nuevo espacio de trabajo de Essentials sin costo adicional. OpenSearch le permite analizar varios orígenes de datos. Incluye espacios de trabajo, que proporcionan experiencias personalizadas para casos de uso populares. Los espacios de trabajo admiten el control de acceso, lo que le permite crear espacios privados para sus casos de uso y compartirlos solo con sus colaboradores.

    b. Use ajustes personalizados:
       i. Elija Personalizar.
       ii. Edite el nombre de la colección y la configuración de retención de datos según sea necesario.
       iii. Seleccione la aplicación OpenSearch y el espacio de trabajo que desea utilizar.
11. Elija Siguiente.
12. Revise sus opciones y seleccione Editar si necesita hacer algún cambio.
13. Seleccione Conector para configurar el origen de datos. Permanezca en esta página mientras se crea el origen de datos. Cuando esté listo, accederá a la página de detalles del origen de datos.

Pasos a seguir a continuación

Visitar OpenSearch Dashboards

Después de crear un origen de datos, OpenSearch Service le proporciona una URL de OpenSearch Dashboards. Se usa para configurar el control de acceso, definir tablas, configurar paneles basados en tipos de registro para los tipos de registro más populares y consultar los datos mediante SQL o PPL.

Para obtener más información, consulte the section called “Configuración de un origen de datos de Registros de CloudWatch”.

Recursos adicionales

Permisos necesarios para los roles de IAM creados manualmente

Al crear un origen de datos, se elige un rol de IAM para administrar el acceso a los datos. Dispone de dos opciones para hacerlo:

1. Crear un nuevo rol de IAM automáticamente
2. Utilizar un rol de IAM creado manualmente

Si utiliza un rol creado manualmente, debe asociar los permisos correctos al rol. Los permisos del rol deben permitir el acceso al origen de datos específico y permitir que OpenSearch Service asuma el rol. Esto es necesario para que OpenSearch Service pueda acceder de forma segura a los datos e interactuar con ellos.

El siguiente ejemplo de política muestra los permisos con privilegios mínimos necesarios para crear y administrar un origen de datos. Si tiene permisos más generales, como `logs:*` o la política de `AdminstratorAccess`, estos engloban los permisos con privilegios mínimos de la política de ejemplo.

En la siguiente política de ejemplo, reemplace el *`texto de marcador`* con su información, según corresponda.

JSON

```
    {
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AmazonOpenSearchDirectQueryAllLogsAccess",
            "Effect": "Allow",
            "Action": [
                "logs:DescribeLogGroups",
                "logs:StartQuery",
                "logs:GetLogGroupFields"
            ],
            "Condition": {
                "StringEquals": {
                    "aws:ResourceAccount": "111122223333"
                }
            },
            "Resource": [
                "arn:aws:logs:us-east-1:111122223333:log-group:*"
            ]
        }
    ]
}
```

El rol debe tener la siguiente política de confianza, que especifica el ID de destino.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "TrustPolicyForAmazonOpenSearchDirectQueryService",
            "Effect": "Allow",
            "Principal": {
                "Service": "directquery.opensearchservice.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "ArnLike": {
                    "aws:SourceArn": "arn:aws:opensearch:us-
east-1:111122223333:datasource/rolename"
```

```
                }
            }
        }
    ]
}
```

Para obtener instrucciones sobre cómo crear los roles, consulte Creación de un rol mediante políticas de confianza personalizadas.

De forma predeterminada, el rol solo tiene acceso a los índices de orígenes de datos de consulta directa. Si bien puede configurar el rol para limitar o conceder el acceso a su origen de datos, se recomienda no ajustar el acceso de este rol. Si elimina el origen de datos, se eliminará este rol. Esto eliminará el acceso de los demás usuarios si están asignados al rol.

Configuración y consulta de un origen de datos de Registros de CloudWatch en OpenSearch Dashboards

Ahora que ha creado su origen de datos, puede comenzar a utilizar OpenSearch Dashboards. En esta sección, se explican varios casos de uso del origen de datos en OpenSearch Dashboards.

Consulte grupos de registros desde la página Discover

En la página OpenSearch Discover, puede utilizar el nuevo origen de datos de consulta directa que configuró para consultar los grupos de registros de Registros de CloudWatch. Para ello, elija Explorar registro y, a continuación, utilice la barra de búsqueda para crear la consulta mediante SQL o PPL. Puede filtrar, ordenar y visualizar los datos devueltos por sus grupos de registros. Para saber qué instrucciones, comandos y limitaciones se admiten en la integración de Registros de CloudWatch, consulte the section called “Comandos SQL y PPL compatibles”.

Crear una vista de panel para el origen de datos

Al utilizar OpenSearch Service, puede analizar rápidamente los tipos de registro de AWS más populares mediante plantillas de panel prediseñadas. Para Registros de CloudWatch, hay plantillas para los registros de VPC, CloudTrail, AWS WAF y Network Firewall. Estas plantillas le permiten crear rápidamente un panel adaptado a sus datos específicos. Incluyen paneles diseñados para ese tipo de registro específico. Esto le permite comenzar a analizar rápidamente estos populares orígenes de registros de AWS, sin tener que compilarlo todo desde cero.

> **Note**
>
> Los Dashboards utilizan vistas indexadas, que ingieren datos de Registros de CloudWatch mediante unidades de cómputo de OpenSearch (OCU) de consulta directa, así como indexingOCU, searchingOCU y almacenamiento de colecciones sin servidor.

Siga estos pasos para crear un panel con una de estas plantillas prediseñadas, de modo que pueda empezar a explorar y analizar sus datos de inmediato.

Cómo crear un panel

1. Vaya a la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/.
2. En el panel de navegación izquierdo, seleccione Administración central y, a continuación, Orígenes de datos conectados.
3. Seleccione el origen de datos para abrir la página de detalles.
4. Elija Crear panel.
5. Elija el tipo de panel que desea crear.
6. Escriba un nombre para su panel de control.
7. Escriba una descripción opcional para el panel de control.
8. Seleccione uno o más grupos de registros para verlos en el panel de control.
9. Elija la frecuencia con la que quiere actualizar los datos del panel.
10. Elija qué espacio de trabajo de OpenSearch desea utilizar.
    a. Para crear un nuevo espacio de trabajo, seleccione Crear nuevo espacio de trabajo y escriba un nombre.
    b. Para usar un espacio de trabajo existente, seleccione Seleccionar espacio de trabajo existente.
11. Elija Crear panel.

Consulta de datos de Registros de CloudWatch en OpenSearch Discover

Para consultar sus datos, seleccione el origen de datos en el menú desplegable. Si utiliza Registros de CloudWatch, vaya a Discover desde el espacio de trabajo de Essentials y comience a consultar datos mediante OpenSearch SQL o el lenguaje de procesamiento de canalizaciones (PPL). Para

obtener una lista de los comandos admitidos, consulte the section called “Comandos SQL y PPL compatibles”.

> **Note**
>
> Si ha creado una vista materializada, puede usar DQL para consultar sus datos, siempre que los haya indexado dentro.

Solución de problemas

Puede haber casos en los que los resultados no devuelvan los resultados esperados. Si tiene algún problema, asegúrese de seguir las the section called “Recomendaciones”.

Consulta directa de datos de Amazon Security Lake en OpenSearch Service

En esta sección, se explica el proceso de creación y configuración de una integración de origen de datos en Amazon OpenSearch Service, lo que le permitirá consultar y analizar de forma eficiente los datos almacenados en Security Lake.

En las siguientes páginas, aprenderá a configurar un origen de datos de consulta directa de Security Lake, conocerá los requisitos previos necesarios y podrá seguir los procedimientos paso a paso con la Consola de administración de AWS.

Temas

- Creación de una integración de origen de datos de Amazon Security Lake en OpenSearch Service
- Configuración y consulta de un origen de datos de Security Lake en OpenSearch Dashboards

Creación de una integración de origen de datos de Amazon Security Lake en OpenSearch Service

Puede utilizar Amazon OpenSearch sin servidor para consultar directamente los datos de seguridad en Amazon Security Lake. Para ello, debe crear un origen de datos que le permita utilizar las capacidades de zero-ETL de OpenSearch en los datos de Security Lake. Al crear un origen de datos, puede buscar directamente los datos almacenados en Security Lake, obtener información sobre ellos y analizarlos. Puede acelerar el rendimiento de sus consultas y utilizar análisis avanzados de

OpenSearch en determinados conjuntos de datos de Security Lake mediante la indexación bajo demanda.

Temas

- Requisitos previos
- Procedimiento
- Pasos a seguir a continuación
- Recursos adicionales

Requisitos previos

Antes de comenzar, asegúrese de que ha revisado la siguiente documentación:

- the section called “Limitaciones de Amazon Security Lake”
- the section called “Recomendaciones para Security Lake”
- the section called “Cuotas para Security Lake”

Antes de poder crear un origen de datos, haga lo siguiente en Security Lake:

- Habilitar Security Hub. Configure Security Lake para recopilar registros en la misma Región de AWS que su recurso de OpenSearch. Para obtener instrucciones, consulte Introducción a Amazon Security Lake en la guía del usuario de Amazon Security Lake.
- Configure los permisos de Security Lake. Asegúrese de haber aceptado los permisos de los roles vinculados al servicio para la administración de recursos y de que la consola no muestre ningún problema en la página Problemas. Para obtener más información, consulte Rol vinculado al servicio para Security Lake en la guía del usuario de Amazon Security Lake.
- Comparta los orígenes de datos de Security Lake. Cuando acceda a OpenSearch con la misma cuenta que Security Lake, asegúrese de que no aparezca ningún mensaje para registrar sus buckets de Security Lake con Lake Formation en la consola de Security Lake. Para acceder a OpenSearch entre cuentas, configure un suscriptor de consultas de Lake Formation en la consola de Security Lake. Use la cuenta asociada a su recurso de OpenSearch como suscriptor. Para obtener más información, consulte Administración de suscriptores en Security Lake en la guía del usuario de Amazon Security Lake.

Además, también debe tener los siguientes recursos en su Cuenta de AWS:

- (Opcional) Un rol de IAM creado manualmente. Puede usar este rol para administrar el acceso al origen de datos. Como alternativa, puede hacer que OpenSearch Service le cree un rol automáticamente con los permisos necesarios. Si decide utilizar un rol de IAM creado manualmente, siga las instrucciones que se indican en the section called “Permisos necesarios para los roles de IAM creados manualmente”.

Procedimiento

Puede configurar un origen de datos para que se conecte a una base de datos de Security Lake desde la Consola de administración de AWS.

Para configurar un origen de datos mediante la Consola de administración de AWS

1. Vaya a la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/.
2. En el panel de navegación de la izquierda, vaya a Administración central y seleccione Orígenes de datos conectados.
3. Elija Conectar.
4. Elija Security Lake como tipo de origen de datos.
5. Elija Siguiente.
6. En Detalles de la conexión de datos, escriba un nombre y una descripción opcional.
7. En Configuración del permiso de acceso de IAM, elija cómo administrar el acceso a su origen de datos.

    a. Si desea crear automáticamente un rol para este origen de datos, siga estos pasos:

        i. Seleccione Crear un nuevo rol.

        ii. Ingrese un nombre para el rol de IAM.

        iii. Seleccione una o más tablas AWS Glue para definir qué datos se pueden consultar.

    b. Si desea utilizar un rol existente que usted administra, siga estos pasos:

        i. Seleccione Usar un rol existente.

        ii. Seleccione un rol existente en el menú desplegable.

> **Note**
>
> Cuando utilice su propio rol, debe asegurarse de que tiene todos los permisos requeridos. Para ello, adjunte las políticas necesarias desde la consola de IAM. Para obtener más información, consulte the section called “Permisos necesarios para los roles de IAM creados manualmente”.

8. (Opcional) En Etiquetas, agregue etiquetas al origen de datos.
9. Elija Siguiente.
10. En Configurar OpenSearch, seleccione cómo configurar OpenSearch.

    - Revise los nombres de los recursos y la configuración de retención de datos predeterminados.

      Al utilizar la configuración predeterminada, se crea una nueva aplicación OpenSearch y un nuevo espacio de trabajo de Essentials sin costo adicional. OpenSearch le permite analizar varios orígenes de datos. Incluye espacios de trabajo, que proporcionan experiencias personalizadas para casos de uso populares. Los espacios de trabajo admiten el control de acceso, lo que le permite crear espacios privados para sus casos de uso y compartirlos solo con sus colaboradores.

11. Use ajustes personalizados:

    a. Elija Personalizar.

    b. Edite el nombre de la colección y la configuración de retención de datos según sea necesario.

    c. Seleccione la aplicación OpenSearch y el espacio de trabajo que desea utilizar.

12. Elija Siguiente.
13. Revise sus opciones y seleccione Editar si necesita hacer algún cambio.
14. Seleccione Conector para configurar el origen de datos. Permanezca en esta página mientras se crea el origen de datos. Cuando esté listo, accederá a la página de detalles del origen de datos.

Pasos a seguir a continuación

Visitar OpenSearch Dashboards y crear un panel

Después de crear un origen de datos, OpenSearch Service le proporciona una URL de OpenSearch Dashboards. Puede utilizar esto para consultar los datos mediante SQL o PPL. La integración de Security Lake incluye plantillas de consulta preempaquetadas para SQL y PPL para que pueda empezar a analizar sus registros.

Para obtener más información, consulte the section called “Configuración de un origen de datos de Security Lake”.

Recursos adicionales

Permisos necesarios para los roles de IAM creados manualmente

Al crear un origen de datos, se elige un rol de IAM para administrar el acceso a los datos. Dispone de dos opciones para hacerlo:

1. Crear un nuevo rol de IAM automáticamente
2. Utilizar un rol de IAM creado manualmente

Si utiliza un rol creado manualmente, debe asociar los permisos correctos al rol. Los permisos del rol deben permitir el acceso al origen de datos específico y permitir que OpenSearch Service asuma el rol de modo que el servicio pueda acceder e interactuar de forma segura con sus datos. Además, conceda permisos a LakeFormation para el rol en las bases de datos y tablas que desee consultar. Otorgue permisos `DESCRIBE` al rol en las bases de datos de SecurityLake que desee consultar desde la conexión de consulta directa. Otorgue al menos los permisos `SELECT and DESCRIBE` al rol de origen de datos para las tablas de la base de datos.

El siguiente ejemplo de política muestra los permisos con privilegios mínimos necesarios para crear y administrar un origen de datos. Si tiene permisos más generales, como la política `AdminstratorAccess`, estos engloban los permisos con privilegios mínimos de la política de ejemplo.

En la siguiente política de ejemplo, reemplace el *`texto de marcador`* con su información, según corresponda.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AmazonOpenSearchDirectQueryServerlessAccess",
            "Effect": "Allow",
            "Action": [
                "aoss:APIAccessAll",
                "aoss:DashboardsAccessAll"
            ],
            "Resource": "arn:aws:aoss:us-
east-1:111122223333:collection/collectionname/*"
        },
        {
            "Sid": "AmazonOpenSearchDirectQueryGlueAccess",
            "Effect": "Allow",
            "Action": [
                "glue:GetDatabase",
                "glue:GetDatabases",
                "glue:GetPartition",
                "glue:GetPartitions",
                "glue:GetTable",
                "glue:GetTableVersions",
                "glue:GetTables",
                "glue:SearchTables",
                "glue:BatchGetPartition"
            ],
            "Resource": [
                "arn:aws:glue:us-east-1:111122223333:table/databasename/*",
                "arn:aws:glue:us-east-1:111122223333:database/databasename",
                "arn:aws:glue:us-east-1:111122223333:catalog",
                "arn:aws:glue:us-east-1:111122223333:database/default"
            ]
        },
        {
            "Sid": "AmazonOpenSearchDirectQueryLakeFormationAccess",
            "Effect": "Allow",
            "Action": [
                "lakeformation:GetDataAccess"
            ],
            "Resource": [
```

```
                "*"
            ]
        }
    ]
}
```

El rol debe tener la siguiente política de confianza, que especifica el ID de destino.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "directquery.opensearchservice.amazonaws.com"
            },
            "Action": "sts:AssumeRole"
        }
    ]
}
```

Para obtener instrucciones sobre cómo crear los roles, consulte Creación de un rol mediante políticas de confianza personalizadas.

De forma predeterminada, el rol solo tiene acceso a los índices de orígenes de datos de consulta directa. Si bien puede configurar el rol para limitar o conceder el acceso a su origen de datos, se recomienda no ajustar el acceso de este rol. Si elimina el origen de datos, se eliminará este rol. Esto eliminará el acceso de los demás usuarios si están asignados al rol.

Consulta de datos de Security Lake que están cifrados con una clave administrada por el cliente

Si el bucket de Security Lake asociado a la conexión de datos se cifra mediante el cifrado del lado del servidor con una AWS KMS key administrada por el cliente, debe agregar el rol de servicio de LakeFormation a la política de claves. Esto permite que el servicio acceda a los datos de sus consultas y los lea.

En la siguiente política de ejemplo, reemplace el *texto de marcador* con su información, según corresponda.

```
{
    "Sid": "Allow LakeFormation to access the key",
    "Effect": "Allow",
    "Principal": {
        "AWS": "arn:aws:iam::account:role/aws-service-role/lakeformation.amazonaws.com/
AWSServiceRoleForLakeFormationDataAccess"
    },
    "Action": [
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:ReEncrypt*",
        "kms:GenerateDataKey*",
        "kms:DescribeKey"
    ],
    "Resource": "*"
}
```

Configuración y consulta de un origen de datos de Security Lake en OpenSearch Dashboards

Ahora que ha creado su origen de datos, puede configurarlo en OpenSearch Dashboards.

En esta sección, se explican varios casos de uso del origen de datos en OpenSearch Dashboards antes de realizar consultas en los datos. Para comenzar, debe ir al origen de datos en OpenSearch Dashboards. En el menú de navegación izquierdo, en Administración, elija Origen de datos. A continuación, seleccione el nombre del origen de datos que creó anteriormente en la consola de OpenSearch Service.

Consulte las tablas de Security Lake de Discover

Si ha creado tablas basadas en sus registros de Security Lake, ahora puede consultarlas directamente desde OpenSearch Discover. Esto le permite acceder y analizar sin problemas los datos almacenados en Security Lake, directamente desde la conocida interfaz Discover. Al consultar Security Lake directamente desde Discover, puede evitar la necesidad de extraer, transformar y cargar manualmente los datos en un índice de búsqueda independiente. Para empezar rápidamente a analizar sus registros, Discover incluye un conjunto de consultas guardadas de PPL y SQL.

Comience por seleccionar el origen de datos que ha configurado. Seleccione la base de datos y la tabla asociadas que desee consultar y, a continuación, utilice la barra de búsqueda para escribir consultas en las tablas. Para saber qué sentencias, comandos y limitaciones son compatibles con la integración de Security Lake, consulte the section called “Comandos SQL y PPL compatibles”.

Para aprovechar las consultas prediseñadas que están disponibles para Security Lake, vaya a ... en la parte superior derecha de Discover, seleccione Abrir consulta y, a continuación, seleccione Plantillas. Hay muchas consultas prediseñadas disponibles para los orígenes de registros compatibles con Security Lake. Busque las plantillas que coincidan con su caso de uso, copie la consulta para utilizarla en la barra de búsqueda y sustituya los campos con plantillas (como región y acción) por su propia información.

Acelere los datos de Discover

Para mejorar el rendimiento y permitir consultas y análisis posteriores más rápidos en OpenSearch, puede ingerir los resultados de la consulta de Discover en una vista indexada de OpenSearch.

Cómo crear una vista indexada

1. En Discover, elija Crear vista indexada.
2. En el editor de consultas, escriba la consulta que prefiera. Puede crear una nueva consulta aquí o utilizar una existente de sus búsquedas anteriores.
3. Especifique un nombre para la nueva vista indexada. Elija un nombre descriptivo que lo ayude a identificar la vista más tarde.
4. Configure los ajustes de retención de datos para la vista indexada. Puede especificar durante cuánto tiempo deben mantenerse los datos en el índice, lo que le permite equilibrar el rendimiento con los costos de almacenamiento.
5. Cree la vista indexada. Una vez creada, la vista indexada estará disponible para realizar consultas y análisis más rápidos.

Si ya ha creado vistas indexadas, puede acceder a ellas desde Discover.

Cómo usar una vista indexada existente

1. En Discover, elija Seleccionar vista indexada para ver una lista de las vistas indexadas existentes de Security Lake.
2. Elija la vista indexada que desea usar. Esto aplicará la vista a la consulta actual, lo que podría acelerar considerablemente la recuperación y el análisis de los datos.

Crear una vista de panel para el origen de datos

Al utilizar OpenSearch Service, puede analizar los tipos de registro de AWS más populares mediante plantillas de panel prediseñadas. Para Security Lake, hay plantillas para registros de VPC, CloudTrail y WAF. Estas plantillas le permiten crear un panel de control adaptado a sus datos específicos. Incluyen consultas y paneles prediseñados que están diseñados para ese tipo de registro específico. Esto le permite comenzar a analizar rápidamente estos populares orígenes de registros de AWS, sin tener que compilarlo todo desde cero.

> **Note**
>
> Los Dashboards utilizan vistas indexadas, que ingieren datos de Security Lake y contribuyen a la computación directa de consultas y colecciones.

Siga estos pasos para crear un panel con una de estas plantillas prediseñadas, de modo que pueda empezar a explorar y analizar sus datos de inmediato.

Cómo crear un panel

1. Vaya a la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/.
2. En el panel de navegación izquierdo, seleccione Administración central y, a continuación, Orígenes de datos conectados.
3. Seleccione el origen de datos para abrir la página de detalles.
4. Elija Crear panel.
5. Elija el tipo de panel que desea crear.
6. Introduzca un nombre para su panel de control.
7. Escriba una descripción opcional para el panel de control.
8. Seleccione una o más tablas de AWS Glue para verlas en su panel de control.
9. Elija la frecuencia con la que quiere actualizar los datos del panel.
10. Elija qué espacio de trabajo de OpenSearch desea utilizar.
    a. Seleccione Crear un espacio de trabajo para crear un espacio de trabajo nuevo.
    b. Para usar un espacio de trabajo existente, seleccione Seleccionar espacio de trabajo existente.
11. Escriba un nombre para su espacio de trabajo.

12. Elija Crear panel.

Solución de problemas

Puede haber casos en los que los resultados no devuelvan los resultados esperados. Si tiene algún problema, asegúrese de seguir las the section called “Recomendaciones”.

Cómo conectar un origen de datos de Amazon OpenSearch Service

La administración del origen de datos es una parte importante del mantenimiento de la fiabilidad, la disponibilidad y el rendimiento de los orígenes de datos de consulta directa y de sus otras soluciones de AWS. AWS ofrece las siguientes herramientas para supervisar, informar cuando algo no va bien y tomar medidas automáticamente cuando proceda.

Temas

- Supervisión con orígenes de datos de métricas de CloudWatch
- Activación y desactivación de orígenes de datos
- Supervisión con AWS Budgets
- Eliminación de un origen de datos

Supervisión con orígenes de datos de métricas de CloudWatch

Puede supervisar la consulta directa mediante CloudWatch. CloudWatch recopila los datos sin procesar y los procesa para convertirlos en métricas legibles casi en tiempo real. Estas estadísticas se mantienen durante 15 meses, de forma que pueda obtener acceso a información histórica y disponer de una mejor perspectiva sobre el rendimiento de su aplicación web o servicio.

También puede establecer alarmas para supervisar determinados umbrales y enviar notificaciones o llevar a cabo acciones cuando se cumplan dichos umbrales. Para obtener más información, consulte What is Amazon CloudWatch.

Amazon S3 notifica las siguientes métricas:

Métrica	Descripción
AsyncQueryCreateAPI	El número total de solicitudes hechas a la API para crear consultas asíncronas. Estadísticas pertinentes: promedio, máximo, suma Dimensiones: `ClientId`,`DomainName` Frecuencia: 60 segundos
AsyncQueryGetApiRequestCount	El número total de solicitudes hechas a la API para recuperar resultados de consultas asíncronas. Estadísticas pertinentes: promedio, máximo, suma Dimensiones: `ClientId`, `DomainName` Frecuencia: 60 segundos
AsyncQueryCancelApiRequestCount	El número total de solicitudes hechas a la API para cancelar consultas asíncronas. Estadísticas pertinentes: promedio, máximo, suma Dimensiones: `ClientId`, `DomainName` Frecuencia: 60 segundos
AsyncQueryGetApiFailedRequestCusErrCount	El número de solicitudes fallidas al recuperar los resultados de una consulta asíncrona debido a errores relacionados con el cliente (por ejemplo, un ID de consulta no válido). Estadísticas pertinentes: promedio, máximo, suma Dimensiones: `ClientId`, `DomainName` Frecuencia: 60 segundos

Métrica	Descripción
AsyncQueryCancelApiFailedRequestCusErrCount	El número de solicitudes fallidas al recuperar los resultados de una consulta asíncrona debido a errores relacionados con el cliente (por ejemplo, un ID de consulta no válido). Estadísticas pertinentes: promedio, máximo, suma Dimensiones: `ClientId`,`DomainName` Frecuencia: 60 segundos
AsyncQueryCancelApiFailedRequestSysErrCount	El número de solicitudes fallidas al crear consultas asíncronas debido a errores relacionados con el cliente. Estadísticas pertinentes: promedio, máximo, suma Dimensiones: `ClientId`, `DomainName` Frecuencia: 60 segundos
AsyncQueryGetApiFailedRequestSysErrCount	El número de solicitudes fallidas al recuperar los resultados de una consulta asíncrona debido a errores relacionados con el sistema. Estadísticas pertinentes: promedio, máximo, suma Dimensiones: `ClientId`, `DomainName` Frecuencia: 60 segundos

Registros de CloudWatch y Security Lake informan las siguientes métricas:

Métrica	Descripción
DirectQueryRate	La tasa de solicitudes realizadas en relación con los orígenes de datos. Estadísticas pertinentes: mínimo, máximo, promedio, suma

Métrica	Descripción
	Dimensiones: `DataSourceName` Frecuencia: 60 segundos
DirectQueryLatency	La latencia observada al ejecutar consultas en los orígenes de datos. Estadísticas pertinentes: mínimo, máximo, promedio, suma, P90, P99 Dimensiones: `DataSourceName` Frecuencia: 60 segundos
FailedDirectQueries	El número total de errores de consulta que se observan en las consultas del origen de datos. Estadísticas pertinentes: mínimo, máximo, promedio, suma Dimensiones: `DataSourceName` Frecuencia: 60 segundos
DirectQueryConsumedOCU	El número de OCU que se consumen para ejecutar las consultas en los orígenes de datos. Estadísticas pertinentes: mínimo, máximo, promedio, suma, P90, P99 Dimensiones: `DataSourceName` Frecuencia: 60 segundos

Activación y desactivación de orígenes de datos

> **Note**
>
> La siguiente información solo se aplica a los orígenes de datos de Amazon S3.

Si desea detener el uso de consultas directas para un origen de datos, puede optar por deshabilitarlo. El deshabilitar un origen de datos, finalizará la ejecución de las consultas existentes e impedirá que el usuario ejecute todas las consultas nuevas.

Las aceleraciones configuradas para mejorar el rendimiento de las consultas, como los índices de omisión, las vistas materializadas o el índice de cobertura, pasarán a ser manuales una vez que se desactive el origen de datos. Una vez que un origen de datos se establece como activo después de deshabilitarlo, las consultas de los usuarios se ejecutarán según lo previsto. Las aceleraciones que anteriormente estaban configuradas como manuales deberán configurarse manualmente para volver a ejecutarse según una programación.

Supervisión con AWS Budgets

Amazon OpenSearch Service está introduciendo los datos de uso de la OCU por cuenta en el Explorador de costos de Administración de facturación y costos. Los clientes pueden contabilizar el uso de las OCU por cuenta y establecer umbrales y alertas cuando estos se superen.

El formato del tipo de uso que se va a filtrar en el Explorador de costos es similar a RegionCode-DirectQueryOCU (OCU/horas). Si desea recibir una notificación cuando el uso de DirectQueryOCU (OCU/horas) alcance su límite, puede crear una cuenta de AWS Budgets y configurar una alerta en función del umbral que haya establecido. De forma opcional, en Amazon S3 se puede optar por configurar un tema de Amazon SNS, que desactivará un origen de datos en caso de que se cumpla un criterio límite.

> **Note**
>
> Los datos de uso de AWS Budgets no están disponibles en tiempo real y pueden demorarse hasta 8 horas.

Eliminación de un origen de datos

Al eliminar un origen de datos, Amazon OpenSearch Service lo elimina de su dominio o su colección. OpenSearch Service también elimina los índices asociados al origen de datos. Sus datos transaccionales no se eliminan del otro Servicio de AWS, pero ese Servicio de AWS no envía datos nuevos a OpenSearch Service.

Puede eliminar una integración de origen de datos mediante la Consola de administración de AWS o la API de OpenSearch Service.

Consola de administración de AWS

Para eliminar un origen de datos de Amazon S3

1. Vaya a la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/.
2. En el panel de navegación izquierdo, seleccione Dominios.
3. Seleccione el dominio del que desea eliminar un origen de datos. Se abrirá la página de detalles del dominio. Elija la pestaña Conexiones que se encuentra debajo de la información general, y busque la sección Consulta directa.
4. Seleccione el origen de datos que desea eliminar, elija Eliminar y confirme la eliminación.

Para eliminar un origen de datos de Registros de CloudWatch o Security Lake

1. Vaya a la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/.
2. En el panel de navegación izquierdo, seleccione Administración central y, a continuación, Orígenes de datos conectados.
3. Seleccione el origen de datos que desea eliminar, elija Eliminar y confirme la eliminación.

API de OpenSearch Service

Para eliminar un origen de datos de Amazon S3, utilice la operación de la API DeleteDataSource.

```
POST https://es.region.amazonaws.com/2021-01-01/opensearch/domain/domain-name/
dataSource/data-source-name
```

Para eliminar un origen de datos de Registros de CloudWatch o Security Lake, utilice la operación de la API DeleteDirectQueryDataSource.

Optimización del rendimiento de las consultas en orígenes de datos de Amazon OpenSearch Service

El rendimiento de las consultas en Amazon OpenSearch Service puede ralentizarse al acceder a orígenes de datos externas. Esto puede deberse a factores como la latencia de la red, la transformación de datos o los grandes volúmenes de datos. Para mejorar el rendimiento, considere la posibilidad de indexar determinadas cantidades de datos según el caso de uso:

- Cómo acelerar las consultas directas en Amazon S3 (índice de saltos)
- Creación de visualizaciones de paneles en Security Lake (vistas materializadas)
- Ingesta de los resultados de las consultas mediante vistas indexadas para revisarlos sin conexión o mejorar el rendimiento en Security Lake (vistas materializadas)

Para obtener la documentación completa sobre las consultas aceleradas, incluidas las consultas de ejemplo, consulte Optimizar el rendimiento de las consultas mediante la indexación de OpenSearch en la documentación de código abierto.

Temas

- Índices de omisión
- Vistas materializadas
- Índices de cobertura

Índices de omisión

Un índice de omisión ingiere solo los metadatos de los datos almacenados en Amazon S3. Cuando consulta una tabla con un índice de omisión, el planificador de consultas hace referencia al índice y vuelve a escribir la consulta para localizar los datos de manera eficiente, en lugar de escanear todas las particiones y archivos. Este enfoque ayuda a delimitar la ubicación exacta de los datos almacenados.

Hay dos formas de crear un índice de omisión. La primera forma consiste en generar automáticamente el índice de omisiones a partir de los detalles del origen de datos. La segunda consiste en utilizar Query Workbench para crear manualmente el índice de saltos mediante una sentencia SQL.

Para generar automáticamente un índice de omisión a partir del origen de datos, vaya a Administración de paneles y a Acelerar datos y, a continuación, seleccione la base de datos y la tabla (puede que tenga que actualizarlas para obtener las bases de datos y tablas más recientes). A continuación, puede elegir Generar para generar automáticamente un índice de omisión o seleccionar manualmente cada campo que desee indexar y especificar la aceleración (tipo de índice de omisión). Por último, elija Crear aceleración para crear una tarea recurrente que rellene el nuevo índice de saltos.

Los índices de omisión solo se admite para orígenes de datos de Amazon S3.

Para obtener más información acerca de cómo configurar índices de omisión mediante Query Workbench, consulte Índices de omisión en la documentación de OpenSearch.

Vistas materializadas

Las vistas materializadas utilizan consultas complejas, como las agregaciones, para admitir las visualizaciones de OpenSearch Search Dashboards. Ingieren un subconjunto de sus datos en función de la consulta y lo almacenan en un índice de OpenSearch. Puede utilizar este índice para crear visualizaciones.

Las vistas materializadas son compatibles con los orígenes de datos de Amazon S3 y Security Lake.

Para obtener más información sobre la configuración de las vistas materializadas mediante Query Workbench, consulte Vistas materializadas en la documentación de OpenSearch.

Índices de cobertura

Un índice de cobertura ingiere datos de una columna especificada en una tabla y OpenSearch crea un nuevo índice basado en estos datos. Puede utilizar este nuevo índice para las visualizaciones y otras características de OpenSearch, como la detección de anomalías o el análisis geoespacial.

Los índices de cobertura solo se admiten para orígenes de datos de Amazon S3.

Para obtener más información acerca de cómo configurar índices de cobertura, consulte Índices de cobertura en la documentación de OpenSearch.

Comandos SQL y PPL compatibles

OpenSearch SQL y OpenSearch Pipeline Processing Language (PPL) son lenguajes para consultar, analizar y procesar datos en OpenSearch, Información de registros de CloudWatch y Security Lake.

Puede utilizar OpenSearch SQL y OpenSearch PPL en OpenSearch Discover para consultar datos en Registros de CloudWatch, Amazon S3 o Security Lake. Información de registros de CloudWatch también es compatible con los lenguajes de consulta OpenSearch PPL y OpenSearch SQL, además de Logs Insights QL, un lenguaje de consultas diseñado específicamente para analizar Registros de CloudWatch.

- OpenSearch SQL: OpenSearch SQL ofrece una opción que es familiar si usted está acostumbrado a trabajar con bases de datos relacionales. OpenSearch SQL ofrece un subconjunto de funciones de SQL, lo que lo convierte en una buena opción para realizar consultas ad hoc y tareas de análisis de datos. Con OpenSearch SQL, puede usar comandos como SELECT, FROM, WHERE, GROUP BY, HAVING y varios otros comandos y funciones de SQL disponibles en SQL. Puede ejecutar JOIN entre tablas (o grupos de registros), correlacionar datos entre tablas (o grupos de registros) mediante subconsultas y utilizar el amplio conjunto de funciones JSON, matemáticas, de cadena, condicionales y otras funciones de SQL para realizar análisis eficaces de los datos de registro y de seguridad.
- OpenSearch PPL (lenguaje de procesamiento de canalizaciones): con OpenSearch PPL, puede recuperar, consultar y analizar datos mediante comandos agrupados, lo que facilita la comprensión y la redacción de consultas complejas. Su sintaxis se basa en las canalizaciones de Unix y permite encadenar comandos para transformar y procesar datos. Con PPL, puede filtrar y agregar datos y utilizar comandos como JOIN, subconsultas, LOOKUP y un amplio conjunto de funciones matemáticas, de cadenas, de fecha, condicionales y de otro tipo para el análisis.

Si bien la mayoría de los comandos de los lenguajes de consulta OpenSearch PPL y OpenSearch SQL son comunes en Registros de CloudWatch y OpenSearch, existen algunas diferencias en cuanto al conjunto de comandos y funciones que se admiten en cada uno de estos servicios. Consulte la siguiente tabla para obtener más detalles.

- the section called “Comandos SQL compatibles”
  - the section called “Información de registros de CloudWatch”
  - the section called “Restricciones SQL generales”
- the section called “Comandos compatibles PPL”
  - the section called “Información adicional para los usuarios de Información de registros de CloudWatch que utilizan OpenSearch PPL”

Comandos y funciones OpenSearch SQL compatibles

En las siguientes tablas de referencia, se muestran qué comandos SQL son compatibles con OpenSearch Discover para consultar datos en Amazon S3, Security Lake o Registros de CloudWatch y qué comandos SQL son compatibles con Información de registros de CloudWatch. La sintaxis SQL admitida en Información de registros de CloudWatch y la admitida en OpenSearch Discover para consultar los Registros de CloudWatch son las mismas y se hace referencia a ellas como Registros de CloudWatch en las siguientes tablas.

> **Note**
>
> OpenSearch también es compatible con SQL para consultar los datos que se ingieren en OpenSearch y se almacenan en índices. Este dialecto de SQL es diferente del SQL utilizado en las consultas directas y se denomina OpenSearch SQL en los índices.

Temas

- Comandos
- Funciones
- Restricciones SQL generales
- Información adicional para los usuarios de Información de registros de CloudWatch que utilizan OpenSearch SQL

Comandos

> **Note**
>
> En la columna de comandos del ejemplo, reemplace *<tableName/logGroup>* según sea necesario en función del origen de datos que esté consultando.
>
> - Comando de ejemplo: `SELECT Body , Operation FROM` **`<tableName/logGroup>`**
> - Si está consultando en Amazon S3 o Security Lake, utilice: `SELECT Body , Operation FROM` **`table_name`**
> - Si está consultando Registros de CloudWatch, utilice: `SELECT Body , Operation FROM` **`` `LogGroupA` ``**

Comando	Descripción	Registros de CloudWatch	Amazon S3	Security Lake	Comando de ejemplo
the section called "Cláusula SELECT"	Muestra los valores proyectados.	compatible	compatible	compatible	`SELECT method, status FROM <tableName/logGroup>`
the section called "Cláusula WHERE"	Filtra los eventos del registro en función de los criterios de campo proporcionados.	compatible	compatible	compatible	`SELECT * FROM <tableName/logGroup> WHERE status = 100`
the section called "Cláusula GROUP BY"	Los grupos registran los eventos según la categoría y encuentran el promedio	compatible	compatible	compatible	`SELECT method, status, COUNT(*) AS request_count, SUM(bytes) AS total_bytes FROM <tableName/logGroup> GROUP BY method, status`

Comando	Descripción	Registros de CloudWatch	Amazon S3	Security Lake	Comando de ejemplo
	en función de las estadísticas.				

the section called “Cláusula HAVING”	Filtra los resultados en función de las condiciones de agrupación.	compatible	compatible	compatible	`SELECT`
					`  method,`
					`  status,`
					`  COUNT(*) AS request_count,`
					`  SUM(bytes) AS total_bytes`
					`FROM`
					`  <tableName/logGroup>`
					`GROUP BY`
					`  method,`
					`  status`
					`HAVING`
					`  COUNT(*) > 5`

Comando	Descripción	Registros de CloudWatch	Amazon S3	Security Lake	Comando de ejemplo
the section called “Cláusula ORDER BY”	Ordena los resultados en función de los campos de la cláusula de orden. Puede especificar un orden tanto ascendente como descendente.	compatible	compatible	compatible	`SELECT * FROM <tableName/logGroup> ORDER BY status DESC`

Comando	Descripción	Registros de CloudWatch	Amazon S3	Security Lake	Comando de ejemplo
the section called “Cláusula JOIN” (INNER | CROSS | LEFT OUTER)	Une los resultados de dos tablas en función de los campos en común.	compatible (se deben usar `Inner` y `Left Outer` como palabras clave para unir; solo se admite una operación JOIN en una sentencia SELECT	compatible (se deben usar las palabras clave Inner, Left Outer y Cross para unir)	compatible (se deben usar las palabras clave Inner, Left Outer y Cross para unir)	`SELECT A.Body, B.Timestamp FROM <tableNameA/logGroupA> AS A INNER JOIN <tableNameB/logGroupB> AS B ON A.`requestId` = B.`requestId``
the section called “Cláusula LIMIT”	Restringe los resultados a las N primeras filas.	compatible	compatible	compatible	`SELECT * FROM <tableName/logGroup> LIMIT 10`

Comando	Descripción	Registros de CloudWatch	Amazon S3	Security Lake	Comando de ejemplo
the section called “Cláusula CASE”	Evalúa las condiciones y devuelve un valor cuando se cumple la primera condición.	compatible	compatible	compatible	`SELECT method, status, CASE WHEN status BETWEEN 100 AND 199 THEN 'Informational' WHEN status BETWEEN 200 AND 299 THEN 'Success' WHEN status BETWEEN 300 AND 399 THEN 'Redirection' WHEN status BETWEEN 400 AND 499 THEN 'Client Error' WHEN status BETWEEN 500 AND 599 THEN 'Server Error' ELSE 'Unknown Status' END AS status_category, CASE method WHEN 'GET' THEN 'Read Operation' WHEN 'POST' THEN 'Create Operation' WHEN 'PUT' THEN 'Update Operation' WHEN 'PATCH' THEN 'Partial Update Operation' WHEN 'DELETE' THEN 'Delete Operation' ELSE 'Other Operation' END AS operation_type, bytes, datetime FROM <tableName/logGroup>`

Comando	Descripción	Registros de CloudWatch	Amazon S3	Security Lake	Comando de ejemplo

the section called “Expresiones de tabla comunes”	Crea un conjunto de resultados temporal con nombre en una sentencia SELECT, INSERT, UPDATE, DELETE o MERGE.	no compatible	compatible	compatible	`WITH RequestStats AS (`
					`  SELECT`
					`  method,`
					`  status,`
					`  bytes,`
					`  COUNT(*) AS request_count`
					`  FROM`
					`  tableName`
					`  GROUP BY`
					`  method,`
					`  status,`
					`  bytes`
					`)`
					`SELECT`
					`  method,`
					`  status,`
					`  bytes,`
					`  request_count`
					`FROM`
					`  RequestStats`
					`WHERE`
					`  bytes > 1000`

the section called “EXPLAIN”	Muestra el plan de ejecución de una sentencia SQL sin ejecutarla.	no compatible	compatible	compatible	`EXPLAIN`
					`SELECT`
					`  k,`
					`  SUM(v)`
					`FROM`
					`  VALUES`
					`  (1, 2),`
					`  (1, 3) AS t(k, v)`
					`GROUP BY`
					`  k`

Comando	Descripción	Registros de CloudWatch	Amazon S3	Security Lake	Comando de ejemplo
the section called “Cláusula LATERAL SUBQUERY”	Permite que una subconsulta de la cláusula FROM haga referencia a las columnas de los elementos anteriores de la misma cláusula FROM.	no compatible	compatible	compatible	`SELECT * FROM tableName LATERAL ( SELECT * FROM t2 WHERE t1.c1 = t2.c1 )`

Comando	Descripción	Registros de CloudWatch	Amazon S3	Security Lake	Comando de ejemplo
the section called “Cláusula LATERAL VIEW”	Genera una tabla virtual mediante la aplicación de una función generadora de tablas a cada fila de una tabla base.	no compatible	compatible	compatible	`SELECT * FROM tableName LATERAL VIEW EXPLODE(ARRAY(30, 60)) tableName AS c_age LATERAL VIEW EXPLODE(ARRAY(40, 80)) AS d_age`
the section called “Predicado LIKE”	Hace coincidir una cadena con un patrón mediante caracteres comodín	compatible	compatible	compatible	`SELECT method, status, request, host FROM <tableName/logGroup> WHERE method LIKE 'D%'`

Comando	Descripción	Registros de CloudWatch	Amazon S3	Security Lake	Comando de ejemplo
the section called "OFFSET"	Especifica el número de filas que se van a omitir antes de empezar a devolver las filas de la consulta	compatible cuando se utiliza junto con una cláusula LIMIT en una consulta. Por ejemplo: • Compare: SELECT * FROM Table LIMIT 100 OFFSET 10 • No compare: SELECT *	compatible	compatible	SELECT method, status, bytes, datetime FROM <tableName/logGroup> ORDER BY datetime OFFSET 10

Comando	Descripción	Registros de CloudWatch	Amazon S3	Security Lake	Comando de ejemplo
		FROM Table OFFSE 10			
the section called “Cláusula PIVOT”	Transfor a las filas en columna rotando los datos de un formato basado en filas a un formato basado en columna	no compatible	compatible	compatible	SELECT * FROM (SELECT method, status, bytes FROM <tableName/logGroup>) AS SourceTable PIVOT (SUM(bytes) FOR method IN ('GET', 'POST', 'PATCH', 'PUT', 'DELETE')) AS PivotTable

Comando	Descripción	Registros de CloudWatch	Amazon S3	Security Lake	Comando de ejemplo
the section called “Operadores de establecimiento”	Combina los resultados de dos o más sentencias SELECT (p. ej., UNION, INTERSE, EXCEPT	compatible	compatible	compatible	`SELECT method, status, bytes FROM <tableName/logGroup> WHERE status = '416' UNION SELECT method, status, bytes FROM <tableName/logGroup> WHERE bytes > 20000`
the section called “Cláusula SORT BY”	Especifica el orden en el que se deben devolver los resultados de la consulta	compatible	compatible	compatible	`SELECT method, status, bytes FROM <tableName/logGroup> SORT BY bytes DESC`

Comando	Descripción	Registros de CloudWatch	Amazon S3	Security Lake	Comando de ejemplo
the section called "UNPIVOT	Transforma las columnas en filas, rotando los datos de un formato basado en columnas a un formato basado en filas.	no compatible	compatible	compatible	`SELECT status, REPLACE(method, '_bytes', '') AS request_method, bytes, datetime FROM PivotedData UNPIVOT ( bytes FOR method IN ( GET_bytes, POST_bytes, PATCH_bytes, PUT_bytes, DELETE_bytes ) ) AS UnpivotedData`

Funciones

> **Note**
>
> En la columna de comandos del ejemplo, reemplace *<tableName/logGroup>* según sea necesario en función del origen de datos que esté consultando.
>
> - Comando de ejemplo: `SELECT Body , Operation FROM` **`<tableName/logGroup>`**
> - Si está consultando en Amazon S3 o Security Lake, utilice: `SELECT Body , Operation FROM` **`table_name`**

- Si está consultando Registros de CloudWatch, utilice: `SELECT Body , Operation FROM `**LogGroupA**``

Gramá SQL disponi e	Descripción	Registr de CloudV h	Amazo S3	Securit Lake	Comando de ejemplo
the section called "Funcic s de cadena	Funciones integradas que pueden manipular y transform ar cadenas y datos de texto en consultas SQL. Por ejemplo, convertir mayúsculas y minúscula s, combinar cadenas, extraer partes y limpiar el texto.	compa e	compa e	compa e	`SELECT UPPER(method) AS upper_method, LOWER(host) AS lower_hos t FROM <tableName/logGroup>`
the section called "Funcic s de fecha	Funciones integrada s para administrar y transform ar los datos	compa e	compa e	compa e	`SELECT TO_TIMESTAMP(datetime) AS timestamp, TIMESTAMP_SECONDS( UNIX_TIMESTAMP(datetime)) AS from_seconds,`

Gramá SQL disponi e	Descripción	Registr de CloudV h	Amazo S3	Securit Lake	Comando de ejemplo
y hora”	de fecha y hora en las consultas. Por ejemplo, date_add, date_form at, datediff y current_d ate.				UNIX_TIMESTAMP(datetime) AS to_unix, FROM_UTC_TIMESTAMP (datetime, 'PST') AS to_pst, TO_UTC_TIMESTAMP(d atetime, 'EST') AS from_est FROM *<tableName/logGroup>*
the section called “Funcic s de agrega n”	Funciones integradas que realizan cálculos en varias filas para generar un único valor resumido. Por ejemplo, suma, recuento, promedio, máximo y mínimo.	compa e	compa e	compa e	SELECT COUNT(*) AS total_rec ords, COUNT(DISTINCT method) AS unique_methods, SUM(bytes) AS total_byt es, AVG(bytes) AS avg_bytes, MIN(bytes) AS min_bytes, MAX(bytes) AS max_bytes FROM *<tableName/logGroup>*

Gramá SQL disponi e	Descripción	Registr de CloudV h	Amazo S3	Securit Lake	Comando de ejemplo
the section called “Funcic s condici ales”	Funciones integradas que realizan acciones en función de condicion es específic as o que evalúan las expresion es de forma condicional. Por ejemplo, CASE e IF.	compa e	compa e	compa e	`SELECT CASE WHEN method = 'GET' AND bytes < 1000 THEN 'Small Read' WHEN method = 'POST' AND bytes > 10000 THEN 'Large Write' WHEN status >= 400 OR bytes = 0 THEN 'Problem' ELSE 'Normal' END AS request_type FROM <tableName/logGroup>`

Gramá SQL disponi e	Descripción	Registr de CloudV h	Amazo S3	Securit Lake	Comando de ejemplo
the section called “Funcic s JSON”	Funciones integrada s para analizar, extraer, modificar y consultar datos con formato JSON en consultas SQL (por ejemplo, from_json , to_json, get_json_ object, json_tupl e) que permiten manipular las estructur as JSON en los conjuntos de datos.	compa e	compa e	compa e	`SELECT FROM_JSON( @message, 'STRUCT< host: STRING, user-identifier: STRING, datetime: STRING, method: STRING, status: INT, bytes: INT >' ) AS parsed_json FROM <tableName/logGroup>`

Gramá SQL disponi e	Descripción	Registr de CloudV h	Amazo S3	Securit Lake	Comando de ejemplo
the section called “Funcic s de matriz”	Funciones integrada s para trabajar con columnas de tipo matriz en consultas SQL, lo que permite realizar operacion es como acceder, modificar y analizar los datos de la matriz (p. ej., size, explode, array_con tains).	compa e	compa e	compa e	`SELECT scores, size(scores) AS length, array_contains(scores, 90) AS has_90 FROM <tableName/logGroup>`

Gramá SQL disponi e	Descripción	Registr de CloudV h	Amazo S3	Securit Lake	Comando de ejemplo
the section called “Funcic s de ventan	Funciones integradas que realizan cálculos en un conjunto específic o de filas relacionadas con la fila (ventana) actual, lo que permite realizar operacion es como la clasificación, los totales acumulado s y los promedios móviles (por ejemplo, ROW_NUMBE R, RANK, LAG, LEAD).	compa e	compa e	compa e	`SELECT field1, field2, RANK() OVER (ORDER BY field2 DESC) AS field2Rank FROM <tableName/logGroup>`

Gramá SQL disponi e	Descripción	Registr de CloudV h	Amazo S3	Securit Lake	Comando de ejemplo
the section called “Funcio s de conver n”	Funciones integrada s para convertir datos de un tipo a otro en las consultas SQL, lo que permite la transform ación de los tipos de datos y las conversio nes de formato (por ejemplo, CAST, TO_DATE, TO_TIMEST AMP, BINARY)	compa e	compa e	compa e	`SELECT CAST('123' AS INT) AS converted_number, CAST(123 AS STRING) AS converted_string FROM <tableName/logGroup>`

Gramá SQL disponi e	Descripción	Registr de CloudV h	Amazo S3	Securit Lake	Comando de ejemplo
the section called “Funcic s de predica s”	Funciones integrada s que evalúan las condiciones y devuelven valores booleanos (verdader o/falso) en función de criterios o patrones específic os (por ejemplo, IN, LIKE, BETWEEN, IS NULL, EXISTS)	compa e	compa e	compa e	`SELECT * FROM <tableName/logGroup> WHERE id BETWEEN 50000 AND 75000`

Gramá SQL disponi e	Descripción	Registr de CloudV h	Amazo S3	Securit Lake	Comando de ejemplo
the section called “Funcic s de mapa”	Aplica una función específic a a cada elemento de una colección y transform a los datos a un nuevo conjunto de valores.	no compat e	compat e	compat e	`SELECT MAP_FILTER( MAP( 'method', method, 'status', CAST(status AS STRING), 'bytes', CAST(bytes AS STRING) ), (k, v) -> k IN ('method', 'status') AND v != 'null' ) AS filtered_map FROM <tableName/logGroup> WHERE status = 100`
the section called “Funcic s matem as”	Realiza operaciones matemátic as con datos numéricos , como calcular promedios , sumas o valores trigonomé tricos.	compat e	compat e	compat e	`SELECT bytes, bytes + 1000 AS added, bytes - 1000 AS subtracted, bytes * 2 AS doubled, bytes / 1024 AS kilobytes, bytes % 1000 AS remainder FROM <tableName/logGroup>`

Gramá SQL disponi e	Descripción	Registr de CloudV h	Amazo S3	Securit Lake	Comando de ejemplo
the section called “Funcic s de grupos de registro múltiple ”	Permite a los usuarios especific ar varios grupos de registros en una sentencia SQL SELECT	compat e	No aplicab	No aplicab	`SELECT lg1.Column1, lg1.Column2 FROM `logGroups(logGrou pIdentifier: ['LogGroup1', 'LogGroup2'])` AS lg1 WHERE lg1.Column3 = "Success"`
the section called “Funcic s genera as”	Crea un objeto iterador que produce una secuencia de valores, lo que permite un uso eficiente de la memoria en conjuntos de datos de gran tamaño.	no compat e	compat e	compat e	`SELECT explode(array(10, 20))`

Restricciones SQL generales

Se aplican las siguientes restricciones al usar OpenSearch SQL con Registros de CloudWatch, Amazon S3 y Security Lake.

1. Solo puede usar una operación JOIN en una sentencia SELECT.
2. Solo se admite un nivel de subconsultas anidadas.
3. No se admiten consultas de varias sentencias separadas por punto y coma.
4. No se admiten consultas que contengan nombres de campo idénticos pero que solo difieran en mayúsculas y minúsculas (como field1 y FIELD1).

   Por ejemplo, no se admiten las siguientes consultas:

   ```
   Select AWSAccountId, awsaccountid from LogGroup
   ```

   Sin embargo, la siguiente consulta se debe a que el nombre del campo (@logStream) es idéntico en ambos grupos de registros:

   ```
   Select a.`@logStream`, b.`@logStream` from Table A INNER Join Table B on a.id = b.id
   ```

5. Las funciones y expresiones deben funcionar con los nombres de los campos y formar parte de una sentencia SELECT con un grupo de registros especificado en la cláusula FROM.

   Por ejemplo, no se admite esta consulta:

   ```
   SELECT cos(10) FROM LogGroup
   ```

   Se admite esta consulta:

   ```
   SELECT cos(field1) FROM LogGroup
   ```

Información adicional para los usuarios de Información de registros de CloudWatch que utilizan OpenSearch SQL

Registros de CloudWatch admite consultas SQL de OpenSearch en la consola, la API y la CLI de Información de registros. Es compatible con la mayoría de los comandos, incluidos SELECT, FROM, WHERE, GROUP BY, HAVING, JOINS y consultas anidadas, además de funciones JSON, matemáticas, de cadena y condicionales. Sin embargo, Registros de CloudWatch solo admite operaciones de lectura, por lo que no permite instrucciones DDL o DML. Consulte las tablas de las secciones anteriores para obtener una lista completa de los comandos y funciones compatibles.

Funciones de grupos de registros múltiples

Información de registros de CloudWatch admite la posibilidad de consultar varios grupos de registros. Para abordar este caso de uso en SQL, puede usar el comando `logGroups`. Este comando es específico para consultar datos en Información de registros de CloudWatch que involucran uno o más grupos de registros. Use esta sintaxis para consultar varios grupos de registros especificándolos en el comando, en lugar de escribir una consulta para cada uno de los grupos de registros y combinarlos con un comando `UNION`.

Sintaxis:

```
`logGroups(
    logGroupIdentifier: ['LogGroup1','LogGroup2', ...'LogGroupn']
)
```

En esta sintaxis, puede especificar hasta 50 grupos de registros en el parámetro `logGroupIndentifier`. Para hacer referencia a los grupos de registros de una cuenta de supervisión, utilice ARN en lugar de nombres `LogGroup`.

Consulta de ejemplo:

```
SELECT LG1.Column1, LG1.Column2 from `logGroups(
    logGroupIdentifier: ['LogGroup1', 'LogGroup2']
)` as LG1
WHERE LG1.Column1 = 'ABC'
```

La siguiente sintaxis que implica varios grupos de registros después de la sentencia `FROM` no se admite al consultar Registros de CloudWatch:

```
SELECT Column1, Column2 FROM 'LogGroup1', 'LogGroup2', ...'LogGroupn'
WHERE Column1 = 'ABC'
```

Restricciones

Cuando utilice comandos SQL o PPL, encierre determinados campos entre comillas invertidas para consultarlos. Las comillas invertidas son obligatorias para los campos con caracteres especiales (no alfabéticos ni numéricos). Por ejemplo, encierre `@message`, `Operation.Export,` y `Test::Field` entre comillas invertidas. No es necesario incluir las columnas con nombres exclusivamente alfabéticos entre comillas invertidas.

Ejemplo de consulta con campos sencillos:

```
SELECT SessionToken, Operation, StartTime  FROM `LogGroup-A`
LIMIT 1000;
```

La misma consulta con comillas invertidas agregadas:

```
SELECT `SessionToken`, `Operation`, `StartTime`  FROM `LogGroup-A`
LIMIT 1000;
```

Para conocer las restricciones generales adicionales que no son específicas de Registros de CloudWatch, consulte the section called “Restricciones SQL generales”.

Consultas y cuotas de ejemplo

> **Note**
>
> Lo siguiente se aplica tanto a los usuarios de Información de registros de CloudWatch como a los usuarios de OpenSearch que consultan datos de CloudWatch.

Para ver ejemplos de consultas SQL que puede usar en Registros de CloudWatch, lea Consultas guardadas y de ejemplo en la consola de Información de registros de Amazon CloudWatch.

Para obtener información acerca de los límites que se aplican al consultar Registros de CloudWatch desde OpenSearch Service, consulte Cuotas de Registros de CloudWatch en la Guía del usuario de Registros de Amazon CloudWatch. Los límites se refieren al número de grupos de Registros de CloudWatch que se pueden consultar, el número máximo de consultas simultáneas que se pueden ejecutar, el tiempo máximo de ejecución de las consultas y el número máximo de filas devueltas en los resultados. Los límites son los mismos independientemente del idioma que utilice para consultar Registros de CloudWatch (es decir, OpenSearch PPL, SQL e Información de registros).

Comandos SQL

Temas

- Funciones de cadena
- Funciones de fecha y hora
- Funciones de agregación

- Funciones condicionales
- Funciones JSON
- Funciones de matriz
- Funciones de ventana
- Funciones de conversión
- Funciones de predicados
- Funciones de mapa
- Funciones matemáticas
- Funciones generadoras
- Cláusula SELECT
- Cláusula WHERE
- Cláusula GROUP BY
- Cláusula HAVING
- Cláusula ORDER BY
- Cláusula JOIN
- Cláusula LIMIT
- Cláusula CASE
- Expresiones de tabla comunes
- EXPLAIN
- Cláusula LATERAL SUBQUERY
- Cláusula LATERAL VIEW
- Predicado LIKE
- OFFSET
- Cláusula PIVOT
- Operadores de establecimiento
- Cláusula SORT BY
- UNPIVOT

Funciones de cadena

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando SQL, consulte the section called “Comandos SQL compatibles”.

Función	Descripción
ascii(str)	Devuelve el valor numérico del primer carácter de `str`.
base64(bin)	Convierte el argumento de binario `bin` a una cadena de base 64.
bit_length(expr)	Devuelve la longitud en bits de los datos de la cadena o el número de bits de los datos binarios.
btrim(str)	Elimina los caracteres de espacio inicial y final de `str`.
btrim(str, trimStr)	Elimine los caracteres `trimStr` iniciales y finales de `str`.
char(expr)	Devuelve el carácter ASCII que tiene el equivalente binario a `expr`. Si n es mayor que 256, el resultado equivale a chr (n % 256)
char_length(expr)	Devuelve la longitud en caracteres de la cadena de datos o el número de bytes de los datos binarios. La longitud de los datos de cadena incluye los espacios finales. La longitud de los datos binarios incluye los ceros binarios.
character_length(expr)	Devuelve la longitud en caracteres de la cadena de datos o el número de bytes de

Función	Descripción
	los datos binarios. La longitud de los datos de cadena incluye los espacios finales. La longitud de los datos binarios incluye los ceros binarios.
chr(expr)	Devuelve el carácter ASCII que tiene el equivalente binario a `expr`. Si n es mayor que 256, el resultado equivale a chr (n % 256)
concat_ws(sep[, str | array(str)]+)	Devuelve la concatenación de las cadenas separadas por `sep`, omitiendo los valores nulos.
contains(left, right)	Devuelve un BOOLEANO. El valor es verdadero si right se encuentra dentro de left. Devuelve NULL si alguna de las expresiones de entrada es NULL. De lo contrario, devuelve que el valor es falso. Tanto left como right deben ser de tipo STRING o BINARY.
decode(bin, charset)	Decodifica el primer argumento utilizando el juego de caracteres del segundo argumento.
decode(expr, search, result [, search, result] ... [, default])	Compara expr con cada valor de búsqueda en orden. Si expr es igual a un valor de búsqueda, decode devuelve el resultado correspondiente. Si no se encuentra coincidencia, devuelve el valor predeterminado. Si se omite el valor predeterminado, devuelve un valor nulo.
elt(n, input1, input2, ...)	Devuelve la n-ésima entrada, por ejemplo, devuelve `input2` cuando n es 2.
encode(str, charset)	Codifica el primer argumento utilizando el conjunto de caracteres del segundo argumento .

Función	Descripción
endswith(left, right)	Devuelve un BOOLEANO. El valor es verdadero si left termina con right. Devuelve NULL si alguna de las expresiones de entrada es NULL. De lo contrario, devuelve que el valor es falso. Tanto left como right deben ser de tipo STRING o BINARY.
find_in_set(str, str_array)	Devuelve el índice (basado en 1) de la cadena dada (`str`) en la lista delimitada por comas (`str_array`). Devuelve 0 si no se encontró la cadena o si la cadena dada (`str`) contiene una coma.
format_number(expr1, expr2)	Formatea el número `expr1` como '#, ###, ##.##', redondeándolo a `expr2` decimales. Si `expr2` es 0, el resultado no tiene separador decimal ni parte fraccionaria. `expr2` también acepta un formato especificado por el usuario. Se supone que funciona como el FORMAT de MySQL.
format_string(strfmt, obj, ...)	Devuelve una cadena formateada a partir de cadenas de formato de estilo printf.
initcap(str)	Devuelve `str` con la primera letra de cada palabra en mayúscula. Todas las demás letras están en minúscula. Las palabras se delimitan mediante espacios en blanco.
instr(str, substr)	Devuelve el índice (basado en 1) de la primera aparición de `substr` en `str`.
lcase(str)	Devuelve `str` con todos los caracteres cambiados a minúsculas.

Función	Descripción
left(str, len)	Devuelve los caracteres más a la izquierda `len` (`len` pueden ser de tipo cadena) de la cadena `str`; si `len` es menor o igual que 0, el resultado es una cadena vacía.
len(expr)	Devuelve la longitud en caracteres de la cadena de datos o el número de bytes de los datos binarios. La longitud de los datos de cadena incluye los espacios finales. La longitud de los datos binarios incluye los ceros binarios.
length(expr)	Devuelve la longitud en caracteres de la cadena de datos o el número de bytes de los datos binarios. La longitud de los datos de cadena incluye los espacios finales. La longitud de los datos binarios incluye los ceros binarios.
levenshtein(str1, str2[, threshold])	Devuelve la distancia de Levenshtein entre las dos cadenas dadas. Si se establece un umbral y la distancia es superior a este, devuelve -1.
locate(substr, str[, pos])	Devuelve la posición de la primera aparición de `substr` en `str` después de la posición de `pos`. El valor `pos` dado y el valor devuelto se basan en 1.
lower(str)	Devuelve `str` con todos los caracteres cambiados a minúsculas.

Función	Descripción
lpad(str, len[, pad])	Devuelve `str`, rellenado a la izquierda con `pad` una longitud de `len`. Si `str` es más larga que `len`, el valor devuelto se acorta al `len` de caracteres de ancho. Si no se especifica `pad`, `str` se rellenará a la izquierda con espacios si es una cadena de caracteres y con ceros si es una secuencia de bytes.
ltrim(str)	Elimina los caracteres de espacio iniciales de `str`.
luhn_check(str)	Comprueba que una cadena de dígitos es válida según el algoritmo de Luhn. Esta función de suma de verificación se aplica en general a los números de tarjetas de crédito y a los números de identificación gubername ntales para distinguir los números válidos de los números mal escritos o incorrectos.
mask(input[, upperChar, lowerChar, digitChar, otherChar])	Enmascara el valor de cadena determinado. La función reemplaza los caracteres por “X” o “x” y los números por “n”. Esto puede resultar útil para crear copias de tablas que tiene información confidencial que ha sido eliminada .
octet_length(expr)	Devuelve la longitud en bytes de los datos de cadena o el número de bytes de los datos binarios.
overlay(input, replace, pos[, len])	Reemplaza `input` por `replace` que comience en `pos` y sea `len` de largo.

Función	Descripción
position(substr, str[, pos])	Devuelve la posición de la primera aparición de `substr` en `str` después de la posición de `pos`. El valor `pos` dado y el valor devuelto se basan en 1.
printf(strfmt, obj, ...)	Devuelve una cadena formateada a partir de cadenas de formato de estilo printf.
regexp_count(str, regexp)	Devuelve un recuento del número de veces que el patrón `regexp` de expresión regular coincide en la cadena `str`.
regexp_extract(str, regexp[, idx])	Extraiga la primera cadena de la `str` que coincida con la `regexp` expresión y que corresponda al índice del grupo de expresion es regulares.
regexp_extract_all(str, regexp[, idx])	Extraiga todas las cadenas que coincidan con la expresión `str` y que correspondan al índice del grupo de expresiones regulares `regexp`.
regexp_instr(str, regexp)	Busca una cadena para un patrón de expresión regular y devuelve un valor entero que indica la posición de inicio o de finalizac ión de la subcadena coincidente. Las posicione s se basan en 1, no en 0. Devuelve 0 si no se encuentran coincidencias.
regexp_replace(str, regexp, rep[, position])	Sustituye todas las subcadenas `str` de esa coincidencia `regexp` por `rep`.
regexp_substr(str, regexp)	Devuelve la subcadena que coincide con la expresión `regexp` regular de la cadena `str`. Si no se encuentra la expresión regular, el resultado también será nulo.

Función	Descripción
repeat(str, n)	Devuelve la cadena que repite el valor de cadena dado n veces.
replace(str, search[, replace])	Sustituye todas las apariciones de `search` por `replace`.
right(str, len)	Devuelve los caracteres más a la derecha `len` (`len` pueden ser de tipo cadena) de la cadena `str`; si `len` es menor o igual que 0, el resultado es una cadena vacía.
rpad(str, len[, pad])	Devuelve `str`, rellenado a la derecha con `pad` una longitud de `len`. Si `str` es más larga que `len`, el valor devuelto se acorta al número de caracteres de `len`. Si `pad` no se especifica, `str` se rellenará a la derecha con espacios si es una cadena de caracteres y con ceros si es una cadena binaria.
rtrim(str)	Elimina los caracteres de espacio del final de `str`.
sentences(str[, lang, country])	`str` se divide en un conjunto de palabras.
soundex(str)	Devuelve el código Soundex de la cadena.
space(n)	Devuelve una cadena que consta de n espacios.
split(str, regex, limit)	Se divide `str` en función de las ocurrencias que coinciden `regex` y devuelve una matriz con una longitud máxima de `limit`

Función	Descripción
split_part(str, delimiter, partNum)	Divide `str` por delimitador y devuelve la parte solicitada de la división (basada en 1). Si alguna entrada es nula, devuelve null. Si `partNum` está fuera del rango de partes divididas, devuelve una cadena vacía. Devuelve error si `partNum` es 0. Si `partNum` es negativo, las partes se cuentan hacia atrás desde el final de la cadena. Si `delimiter` es una cadena vacía, `str` no está dividida.
startswith(left, right)	Devuelve un BOOLEANO. El valor es verdadero si left comienza con right. Devuelve NULL si alguna de las expresiones de entrada es NULL. De lo contrario, devuelve que el valor es falso. Tanto left como right deben ser de tipo STRING o BINARY.
substr(str, pos[, len])	Devuelve la subcadena `str` que comienza en `pos` y es de longitud `len`, o el segmento de la matriz de bytes que comienza en `pos` y es de longitud `len`.
substr(str FROM pos[FOR len]])	Devuelve la subcadena `str` que comienza en `pos` y es de longitud `len`, o el segmento de la matriz de bytes que comienza en `pos` y es de longitud `len`.
substring(str, pos[, len])	Devuelve la subcadena `str` que comienza en `pos` y es de longitud `len`, o el segmento de la matriz de bytes que comienza en `pos` y es de longitud `len`.

Función	Descripción
substring(str FROM pos[FOR len]])	Devuelve la subcadena `str` que comienza en `pos` y es de longitud `len`, o el segmento de la matriz de bytes que comienza en `pos` y es de longitud `len`.
substring_index(str, delim, count)	Devuelve la subcadena `str` antes de la aparición `count` del delimitador `delim`. Si `count` es positivo, devuelve todo lo que esté a la izquierda del delimitador final (contando desde la izquierda). Si `count` es negativo, devuelve todo lo que esté a la derecha del delimitador final (contando desde la derecha). La función substring_index realiza una búsqueda de `delim` que distingue entre mayúsculas y minúsculas.
to_binary(str[, fmt])	Convierte la entrada `str` en un valor binario en función del `fmt` proporcionado. `fmt` puede ser un literal de cadena “hex”, “utf-8”, “utf8” o “base64” que no distinga mayúsculas de minúsculas. De forma predeterminada, el formato binario para la conversión es “hex” si se omite el `fmt`. La función devuelve NULL si al menos uno de los parámetros de entrada es NULL.

Función	Descripción
to_char(numberExpr, formatExpr)	Convierte `numberExpr` en una cadena basada en `formatExpr`. Genera una excepción si no se cumple ninguna de las condiciones. El formato puede constar de los siguientes caracteres, sin distinguir entre mayúsculas y minúsculas: “0” o “9”: especifica un dígito esperado entre 0 y 9. Una secuencia de 0 o 9 en la cadena de formato coincide con una secuencia de dígitos del valor de entrada, lo que genera una cadena de resultado s de la misma longitud que la secuencia correspondiente de la cadena de formato. La cadena resultante se rellena con ceros a la izquierda si la secuencia 0/9 incluye más dígitos que la parte correspondiente del valor decimal, comienza por 0 y está antes de la coma decimal. De lo contrario, se rellena con espacios. “.” o “D”: especifica la posición de la coma decimal (opcional, solo se permite una vez). “,” o “G”: especifica la posición del separador de agrupamiento (miles) (,). Debe haber un 0 o un 9 a la izquierda y a la derecha de cada separador de agrupamiento. '

Función	Descripción
tto_number(expr, fmt)	Convierte la cadena 'expr' en un número según el formato de cadena 'fmt'. Genera una excepción si no se cumple ninguna de las condiciones. El formato puede constar de los siguientes caracteres, sin distinguir entre mayúsculas y minúsculas: “0” o “9”: especifica un dígito esperado entre 0 y 9. Una secuencia de 0 o 9 en la cadena de formato coincide con una secuencia de dígitos de la cadena de entrada. Si la secuencia 0/9 comienza por 0 y está antes de la coma decimal, solo puede coincidir con una secuencia de dígitos del mismo tamaño. De lo contrario, si la secuencia comienza con 9 o está después de la coma decimal, puede coincidir con una secuencia de dígitos que tenga el mismo tamaño o menor. “.” o “D”: especifica la posición de la coma decimal (opcional, solo se permite una vez). “,” o “G”: especifica la posición del separador de agrupamiento (miles) (,). Debe haber un 0 o un 9 a la izquierda y a la derecha de cada separador de agrupamiento. “expr” debe coincidir con el separador de agrupamiento correspondiente al tamaño del número. '

Función	Descripción
to_varchar(numberExpr, formatExpr)	Convierte `numberExpr` en una cadena basada en `formatExpr`. Genera una excepción si no se cumple ninguna de las condiciones. El formato puede constar de los siguientes caracteres, sin distinguir entre mayúsculas y minúsculas: “0” o “9”: especifica un dígito esperado entre 0 y 9. Una secuencia de 0 o 9 en la cadena de formato coincide con una secuencia de dígitos del valor de entrada, lo que genera una cadena de resultado s de la misma longitud que la secuencia correspondiente de la cadena de formato. La cadena resultante se rellena con ceros a la izquierda si la secuencia 0/9 incluye más dígitos que la parte correspondiente del valor decimal, comienza por 0 y está antes de la coma decimal. De lo contrario, se rellena con espacios. “.” o “D”: especifica la posición de la coma decimal (opcional, solo se permite una vez). “,” o “G”: especifica la posición del separador de agrupamiento (miles) (,). Debe haber un 0 o un 9 a la izquierda y a la derecha de cada separador de agrupamiento. '
translate(input, from, to)	Traduce la cadena `input` y los caracteres encontrados en la cadena `from` reemplaza dos por los caracteres correspondientes en la cadena `to`.
trim(str)	Elimina los caracteres de espacio inicial y final de `str`.
trim(BOTH FROM str)	Elimina los caracteres de espacio inicial y final de `str`.

Función	Descripción
trim(LEADING FROM str)	Elimina los caracteres de espacio iniciales de `str`.
trim(TRAILING FROM str)	Elimina los caracteres de espacio del final de `str`.
trim(trimStr FROM str)	Elimine los caracteres `trimStr` iniciales y finales de `str`.
trim(BOTH trimStr FROM str)	Elimine los caracteres `trimStr` iniciales y finales de `str`.
trim(LEADING trimStr FROM str)	Elimine los caracteres `trimStr` iniciales de `str`.
trim(TRAILING trimStr FROM str)	Elimine los caracteres `trimStr` finales de `str`.
try_to_binary(str[, fmt])	Se trata de una versión de `to_binary` especial que realiza la misma operación, pero devuelve un valor NULL en lugar de generar un error si no se puede realizar la conversión.
try_to_number(expr, fmt)	Convierte la cadena 'expr' en un número según el formato de la cadena `fmt`. Devuelve NULL si la cadena 'expr' no coincide con el formato esperado. El formato sigue la misma semántica que la función to_number.
ucase(str)	Devuelve `str` con todos los caracteres cambiados a mayúsculas.
unbase64(str)	Convierte el argumento de una cadena `str` de base 64 en un binario.
upper(str)	Devuelve `str` con todos los caracteres cambiados a mayúsculas.

Ejemplos

```
-- ascii
SELECT ascii('222');
+----------+
|ascii(222)|
+----------+
|        50|
+----------+
SELECT ascii(2);
+--------+
|ascii(2)|
+--------+
|      50|
+--------+
-- base64
SELECT base64('Feathers');
+-----------------+
|base64(Feathers)|
+-----------------+
|     RmVhdGhlcnM=|
+-----------------+
SELECT base64(x'537061726b2053514c');
+-----------------------------+
|base64(X'537061726B2053514C')|
+-----------------------------+
|                 U3BhcmsgU1FM|
+-----------------------------+
-- bit_length
SELECT bit_length('Feathers');
+---------------------+
|bit_length(Feathers)|
+---------------------+
|                   64|
+---------------------+
SELECT bit_length(x'537061726b2053514c');
+---------------------------------+
|bit_length(X'537061726B2053514C')|
+---------------------------------+
|                               72|
+---------------------------------+
-- btrim
SELECT btrim('    Feathers   ');
```

```
+----------------------+
|btrim(    Feathers   )|
+----------------------+
|              Feathers|
+----------------------+
SELECT btrim(encode('    Feathers   ', 'utf-8'));
+-----------------------------------+
|btrim(encode(    Feathers   , utf-8))|
+-----------------------------------+
|                           Feathers|
+-----------------------------------+
SELECT btrim('Feathers', 'Fe');
+---------------------+
|btrim(Alphabet, Al)|
+---------------------+
|               athers|
+---------------------+
SELECT btrim(encode('Feathers', 'utf-8'), encode('Al', 'utf-8'));
+-----------------------------------------------------+
|btrim(encode(Feathers, utf-8), encode(Al, utf-8))|
+-----------------------------------------------------+
|                                               athers|
+-----------------------------------------------------+
-- char
SELECT char(65);
+--------+
|char(65)|
+--------+
|       A|
+--------+
-- char_length
SELECT char_length('Feathers ');
+-----------------------+
|char_length(Feathers )|
+-----------------------+
|                     9 |
+-----------------------+
SELECT char_length(x'537061726b2053514c');
+---------------------------------+
|char_length(X'537061726B2053514C')|
+---------------------------------+
|                                9|
+---------------------------------+
SELECT CHAR_LENGTH('Feathers ');
```

```
+----------------------+
|char_length(Feathers )|
+----------------------+
|                     9|
+----------------------+
SELECT CHARACTER_LENGTH('Feathers ');
+---------------------------+
|character_length(Feathers )|
+---------------------------+
|                          9|
+---------------------------+
-- character_length
SELECT character_length('Feathers ');
+---------------------------+
|character_length(Feathers )|
+---------------------------+
|                          9|
+---------------------------+
SELECT character_length(x'537061726b2053514c');
+--------------------------------------+
|character_length(X'537061726B2053514C')|
+--------------------------------------+
|                                     9|
+--------------------------------------+
SELECT CHAR_LENGTH('Feathers ');
+----------------------+
|char_length(Feathers )|
+----------------------+
|                     9|
+----------------------+
SELECT CHARACTER_LENGTH('Feathers ');
+---------------------------+
|character_length(Feathers )|
+---------------------------+
|                          9|
+---------------------------+
-- chr
SELECT chr(65);
+-------+
|chr(65)|
+-------+
|      A|
+-------+
-- concat_ws
```

```
SELECT concat_ws(' ', 'Fea', 'thers');
+-----------------------+
|concat_ws( , Fea, thers)|
+-----------------------+
|               Feathers|
+-----------------------+
SELECT concat_ws('s');
+------------+
|concat_ws(s)|
+------------+
|            |
+------------+
SELECT concat_ws('/', 'foo', null, 'bar');
+---------------------------+
|concat_ws(/, foo, NULL, bar)|
+---------------------------+
|                    foo/bar|
+---------------------------+
SELECT concat_ws(null, 'Fea', 'thers');
+---------------------------+
|concat_ws(NULL, Fea, thers)|
+---------------------------+
|                       NULL|
+---------------------------+
-- contains
SELECT contains('Feathers', 'Fea');
+-------------------------+
|contains(Feathers, Fea)|
+-------------------------+
|                     true|
+-------------------------+
SELECT contains('Feathers', 'SQL');
+-------------------------+
|contains(Feathers, SQL)|
+-------------------------+
|                    false|
+-------------------------+
SELECT contains('Feathers', null);
+-------------------------+
|contains(Feathers, NULL)|
+-------------------------+
|                    NULL|
+-------------------------+
SELECT contains(x'537061726b2053514c', x'537061726b');
```

```
+-----------------------------------------------+
|contains(X'537061726B2053514C', X'537061726B')|
+-----------------------------------------------+
|                                           true|
+-----------------------------------------------+
-- decode
SELECT decode(encode('abc', 'utf-8'), 'utf-8');
+---------------------------------+
|decode(encode(abc, utf-8), utf-8)|
+---------------------------------+
|                              abc|
+---------------------------------+
SELECT decode(2, 1, 'Southlake', 2, 'San Francisco', 3, 'New Jersey', 4, 'Seattle',
 'Non domestic');
+-------------------------------------------------------------------------------------+
|decode(2, 1, Southlake, 2, San Francisco, 3, New Jersey, 4, Seattle, Non domestic)|
+-------------------------------------------------------------------------------------+
|                                                                        San Francisco|
+-------------------------------------------------------------------------------------+
SELECT decode(6, 1, 'Southlake', 2, 'San Francisco', 3, 'New Jersey', 4, 'Seattle',
 'Non domestic');
+-------------------------------------------------------------------------------------+
|decode(6, 1, Southlake, 2, San Francisco, 3, New Jersey, 4, Seattle, Non domestic)|
+-------------------------------------------------------------------------------------+
|                                                                          Non domestic|
+-------------------------------------------------------------------------------------+
SELECT decode(6, 1, 'Southlake', 2, 'San Francisco', 3, 'New Jersey', 4, 'Seattle');
+-----------------------------------------------------------------+
|decode(6, 1, Southlake, 2, San Francisco, 3, New Jersey, 4, Seattle)|
+-----------------------------------------------------------------+
|                                                             NULL|
+-----------------------------------------------------------------+
SELECT decode(null, 6, 'Fea', NULL, 'thers', 4, 'rock');
+----------------------------------------+
|decode(NULL, 6, Fea, NULL, thers, 4, rock)|
+----------------------------------------+
|                                   thers|
+----------------------------------------+
-- elt
SELECT elt(1, 'scala', 'java');
+------------------+
|elt(1, scala, java)|
+------------------+
|             scala|
```

```
+-------------------+
SELECT elt(2, 'a', 1);
+------------+
|elt(2, a, 1)|
+------------+
|           1|
+------------+
-- encode
SELECT encode('abc', 'utf-8');
+------------------+
|encode(abc, utf-8)|
+------------------+
|        [61 62 63]|
+------------------+
-- endswith
SELECT endswith('Feathers', 'ers');
+------------------------+
|endswith(Feathers, ers)|
+------------------------+
|                    true|
+------------------------+
SELECT endswith('Feathers', 'SQL');
+-------------------------+
|endswith(Feathers, SQL)|
+-------------------------+
|                     false|
+-------------------------+
SELECT endswith('Feathers', null);
+-------------------------+
|endswith(Feathers, NULL)|
+-------------------------+
|                     NULL|
+-------------------------+
SELECT endswith(x'537061726b2053514c', x'537061726b');
+-----------------------------------------------+
|endswith(X'537061726B2053514C', X'537061726B')|
+-----------------------------------------------+
|                                          false|
+-----------------------------------------------+
SELECT endswith(x'537061726b2053514c', x'53514c');
+---------------------------------------+
|endswith(X'537061726B2053514C', X'53514C')|
+---------------------------------------+
|                                   true|
```

```
+-----------------------------------------+
-- find_in_set
SELECT find_in_set('ab','abc,b,ab,c,def');
+-------------------------------+
|find_in_set(ab, abc,b,ab,c,def)|
+-------------------------------+
|                              3|
+-------------------------------+
-- format_number
SELECT format_number(12332.123456, 4);
+-----------------------------+
|format_number(12332.123456, 4)|
+-----------------------------+
|                  12,332.1235|
+-----------------------------+
SELECT format_number(12332.123456, '##################.###');
+---------------------------------------------------+
|format_number(12332.123456, ##################.###)|
+---------------------------------------------------+
|                                           12332.123|
+---------------------------------------------------+
-- format_string
SELECT format_string("Hello World %d %s", 100, "days");
+-----------------------------------------+
|format_string(Hello World %d %s, 100, days)|
+-----------------------------------------+
|                      Hello World 100 days|
+-----------------------------------------+
-- initcap
SELECT initcap('Feathers');
+------------------+
|initcap(Feathers)|
+------------------+
|         Feathers|
+------------------+
-- instr
SELECT instr('Feathers', 'ers');
+--------------------+
|instr(Feathers, ers)|
+--------------------+
|                   6|
+--------------------+
-- lcase
SELECT lcase('Feathers');
```

```
+---------------+
|lcase(Feathers)|
+---------------+
|       feathers|
+---------------+
-- left
SELECT left('Feathers', 3);
+------------------+
|left(Feathers, 3)|
+------------------+
|               Fea|
+------------------+
SELECT left(encode('Feathers', 'utf-8'), 3);
+---------------------------------+
|left(encode(Feathers, utf-8), 3)|
+---------------------------------+
|                       [RmVh]|
+---------------------------------+
-- len
SELECT len('Feathers ');
+---------------+
|len(Feathers )|
+---------------+
|             9|
+---------------+
SELECT len(x'537061726b2053514c');
+--------------------------+
|len(X'537061726B2053514C')|
+--------------------------+
|                         9|
+--------------------------+
SELECT CHAR_LENGTH('Feathers ');
+-----------------------+
|char_length(Feathers )|
+-----------------------+
|                      9|
+-----------------------+
SELECT CHARACTER_LENGTH('Feathers ');
+----------------------------+
|character_length(Feathers )|
+----------------------------+
|                           9|
+----------------------------+
-- length
```

```
SELECT length('Feathers ');
+-----------------+
|length(Feathers )|
+-----------------+
|               9|
+-----------------+
SELECT length(x'537061726b2053514c');
+----------------------------+
|length(X'537061726B2053514C')|
+----------------------------+
|                           9|
+----------------------------+
SELECT CHAR_LENGTH('Feathers ');
+----------------------+
|char_length(Feathers )|
+----------------------+
|                     9|
+----------------------+
SELECT CHARACTER_LENGTH('Feathers ');
+---------------------------+
|character_length(Feathers )|
+---------------------------+
|                          9|
+---------------------------+
-- levenshtein
SELECT levenshtein('kitten', 'sitting');
+---------------------------+
|levenshtein(kitten, sitting)|
+---------------------------+
|                           3|
+---------------------------+
SELECT levenshtein('kitten', 'sitting', 2);
+-------------------------------+
|levenshtein(kitten, sitting, 2)|
+-------------------------------+
|                             -1|
+-------------------------------+
-- locate
SELECT locate('bar', 'foobarbar');
+-------------------------+
|locate(bar, foobarbar, 1)|
+-------------------------+
|                        4|
+-------------------------+
```

```
SELECT locate('bar', 'foobarbar', 5);
+-------------------------+
|locate(bar, foobarbar, 5)|
+-------------------------+
|                        7|
+-------------------------+
SELECT POSITION('bar' IN 'foobarbar');
+-------------------------+
|locate(bar, foobarbar, 1)|
+-------------------------+
|                        4|
+-------------------------+
-- lower
SELECT lower('Feathers');
+---------------+
|lower(Feathers)|
+---------------+
|       feathers|
+---------------+
-- lpad
SELECT lpad('hi', 5, '??');
+---------------+
|lpad(hi, 5, ??)|
+---------------+
|          ???hi|
+---------------+
SELECT lpad('hi', 1, '??');
+---------------+
|lpad(hi, 1, ??)|
+---------------+
|              h|
+---------------+
SELECT lpad('hi', 5);
+--------------+
|lpad(hi, 5,  )|
+--------------+
|            hi|
+--------------+
SELECT hex(lpad(unhex('aabb'), 5));
+--------------------------------+
|hex(lpad(unhex(aabb), 5, X'00'))|
+--------------------------------+
|                     000000AABB|
+--------------------------------+
```

```
SELECT hex(lpad(unhex('aabb'), 5, unhex('1122')));
+-------------------------------------+
|hex(lpad(unhex(aabb), 5, unhex(1122)))|
+-------------------------------------+
|                           112211AABB|
+-------------------------------------+
-- ltrim
SELECT ltrim('    Feathers   ');
+----------------------+
|ltrim(    Feathers   )|
+----------------------+
|           Feathers   |
+----------------------+
-- luhn_check
SELECT luhn_check('8112189876');
+----------------------+
|luhn_check(8112189876)|
+----------------------+
|                  true|
+----------------------+
SELECT luhn_check('79927398713');
+-----------------------+
|luhn_check(79927398713)|
+-----------------------+
|                   true|
+-----------------------+
SELECT luhn_check('79927398714');
+-----------------------+
|luhn_check(79927398714)|
+-----------------------+
|                  false|
+-----------------------+
-- mask
SELECT mask('abcd-EFGH-8765-4321');
+---------------------------------------+
|mask(abcd-EFGH-8765-4321, X, x, n, NULL)|
+---------------------------------------+
|                    xxxx-XXXX-nnnn-nnnn|
+---------------------------------------+
SELECT mask('abcd-EFGH-8765-4321', 'Q');
+---------------------------------------+
|mask(abcd-EFGH-8765-4321, Q, x, n, NULL)|
+---------------------------------------+
|                    xxxx-QQQQ-nnnn-nnnn|
```

```
+-----------------------------------------+
SELECT mask('AbCD123-@$#', 'Q', 'q');
+---------------------------------+
|mask(AbCD123-@$#, Q, q, n, NULL)|
+---------------------------------+
|                     QqQQnnn-@$#|
+---------------------------------+
SELECT mask('AbCD123-@$#');
+---------------------------------+
|mask(AbCD123-@$#, X, x, n, NULL)|
+---------------------------------+
|                     XxXXnnn-@$#|
+---------------------------------+
SELECT mask('AbCD123-@$#', 'Q');
+---------------------------------+
|mask(AbCD123-@$#, Q, x, n, NULL)|
+---------------------------------+
|                     QxQQnnn-@$#|
+---------------------------------+
SELECT mask('AbCD123-@$#', 'Q', 'q');
+---------------------------------+
|mask(AbCD123-@$#, Q, q, n, NULL)|
+---------------------------------+
|                     QqQQnnn-@$#|
+---------------------------------+
SELECT mask('AbCD123-@$#', 'Q', 'q', 'd');
+---------------------------------+
|mask(AbCD123-@$#, Q, q, d, NULL)|
+---------------------------------+
|                     QqQQddd-@$#|
+---------------------------------+
SELECT mask('AbCD123-@$#', 'Q', 'q', 'd', 'o');
+-----------------------------+
|mask(AbCD123-@$#, Q, q, d, o)|
+-----------------------------+
|                  QqQQdddoooo|
+-----------------------------+
SELECT mask('AbCD123-@$#', NULL, 'q', 'd', 'o');
+---------------------------------+
|mask(AbCD123-@$#, NULL, q, d, o)|
+---------------------------------+
|                     AqCDdddoooo|
+---------------------------------+
SELECT mask('AbCD123-@$#', NULL, NULL, 'd', 'o');
```

```
+----------------------------------+
|mask(AbCD123-@$#, NULL, NULL, d, o)|
+----------------------------------+
|                       AbCDdddoooo|
+----------------------------------+
SELECT mask('AbCD123-@$#', NULL, NULL, NULL, 'o');
+-------------------------------------+
|mask(AbCD123-@$#, NULL, NULL, NULL, o)|
+-------------------------------------+
|                           AbCD123oooo|
+-------------------------------------+
SELECT mask(NULL, NULL, NULL, NULL, 'o');
+-------------------------------+
|mask(NULL, NULL, NULL, NULL, o)|
+-------------------------------+
|                           NULL|
+-------------------------------+
SELECT mask(NULL);
+-------------------------+
|mask(NULL, X, x, n, NULL)|
+-------------------------+
|                     NULL|
+-------------------------+
SELECT mask('AbCD123-@$#', NULL, NULL, NULL, NULL);
+----------------------------------------+
|mask(AbCD123-@$#, NULL, NULL, NULL, NULL)|
+----------------------------------------+
|                             AbCD123-@$#|
+----------------------------------------+
-- octet_length
SELECT octet_length('Feathers');
+----------------------+
|octet_length(Feathers)|
+----------------------+
|                     8|
+----------------------+
SELECT octet_length(x'537061726b2053514c');
+----------------------------------+
|octet_length(X'537061726B2053514C')|
+----------------------------------+
|                                 9|
+----------------------------------+
-- overlay
SELECT overlay('Feathers' PLACING '_' FROM 6);
```

```
+---------------------------+
|overlay(Feathers, _, 6, -1)|
+---------------------------+
|                  Feathe_ers|
+---------------------------+
SELECT overlay('Feathers' PLACING 'ures' FROM 5);
+-----------------------------+
|overlay(Feathers, ures, 5, -1)|
+-----------------------------+
|                     Features  |
+-----------------------------+
-- position
SELECT position('bar', 'foobarbar');
+---------------------------+
|position(bar, foobarbar, 1)|
+---------------------------+
|                          4|
+---------------------------+
SELECT position('bar', 'foobarbar', 5);
+---------------------------+
|position(bar, foobarbar, 5)|
+---------------------------+
|                          7|
+---------------------------+
SELECT POSITION('bar' IN 'foobarbar');
+-------------------------+
|locate(bar, foobarbar, 1)|
+-------------------------+
|                        4|
+-------------------------+
-- printf
SELECT printf("Hello World %d %s", 100, "days");
+-----------------------------------+
|printf(Hello World %d %s, 100, days)|
+-----------------------------------+
|               Hello World 100 days|
+-----------------------------------+
-- regexp_count
SELECT regexp_count('Steven Jones and Stephen Smith are the best players', 'Ste(v|
ph)en');
+-------------------------------------------------------------------------------+
|regexp_count(Steven Jones and Stephen Smith are the best players, Ste(v|ph)en)|
+-------------------------------------------------------------------------------+
|                                                                              2|
```

```
+--------------------------------------------------------------------------------+
SELECT regexp_count('abcdefghijklmnopqrstuvwxyz', '[a-z]{3}');
+-------------------------------------------------+
|regexp_count(abcdefghijklmnopqrstuvwxyz, [a-z]{3})|
+-------------------------------------------------+
|                                                8|
+-------------------------------------------------+
-- regexp_extract
SELECT regexp_extract('100-200', '(\\d+)-(\\d+)', 1);
+--------------------------------------+
|regexp_extract(100-200, (\d+)-(\d+), 1)|
+--------------------------------------+
|                                   100|
+--------------------------------------+
-- regexp_extract_all
SELECT regexp_extract_all('100-200, 300-400', '(\\d+)-(\\d+)', 1);
+-----------------------------------------------------+
|regexp_extract_all(100-200, 300-400, (\d+)-(\d+), 1)|
+-----------------------------------------------------+
|                                           [100, 300]|
+-----------------------------------------------------+
-- regexp_instr
SELECT regexp_instr('user@opensearch.org', '@[^.]*');
+-----------------------------------------------+
|regexp_instr(user@opensearch.org, @[^.]*, 0)|
+-----------------------------------------------+
|                                              5|
+-----------------------------------------------+
-- regexp_replace
SELECT regexp_replace('100-200', '(\\d+)', 'num');
+--------------------------------------+
|regexp_replace(100-200, (\d+), num, 1)|
+--------------------------------------+
|                               num-num|
+--------------------------------------+
-- regexp_substr
SELECT regexp_substr('Steven Jones and Stephen Smith are the best players', 'Ste(v|
ph)en');
+-----------------------------------------------------------------------------+
|regexp_substr(Steven Jones and Stephen Smith are the best players, Ste(v|ph)en)|
+-----------------------------------------------------------------------------+
|                                                                       Steven|
+-----------------------------------------------------------------------------+
SELECT regexp_substr('Steven Jones and Stephen Smith are the best players', 'Jeck');
```

```
+-------------------------------------------------------------------------+
|regexp_substr(Steven Jones and Stephen Smith are the best players, Jeck)|
+-------------------------------------------------------------------------+
|                                                                     NULL|
+-------------------------------------------------------------------------+
-- repeat
SELECT repeat('123', 2);
+--------------+
|repeat(123, 2)|
+--------------+
|        123123|
+--------------+
-- replace
SELECT replace('ABCabc', 'abc', 'DEF');
+-------------------------+
|replace(ABCabc, abc, DEF)|
+-------------------------+
|                   ABCDEF|
+-------------------------+
-- right
SELECT right('Feathers', 3);
+-------------------+
|right(Feathers, 3)|
+-------------------+
|                ers|
+-------------------+
-- rpad
SELECT rpad('hi', 5, '??');
+---------------+
|rpad(hi, 5, ??)|
+---------------+
|          hi???|
+---------------+
SELECT rpad('hi', 1, '??');
+---------------+
|rpad(hi, 1, ??)|
+---------------+
|              h|
+---------------+
SELECT rpad('hi', 5);
+--------------+
|rpad(hi, 5,  )|
+--------------+
|         hi   |
```

```
+--------------+
SELECT hex(rpad(unhex('aabb'), 5));
+--------------------------------+
|hex(rpad(unhex(aabb), 5, X'00'))|
+--------------------------------+
|                      AABB000000|
+--------------------------------+
SELECT hex(rpad(unhex('aabb'), 5, unhex('1122')));
+-------------------------------------+
|hex(rpad(unhex(aabb), 5, unhex(1122)))|
+-------------------------------------+
|                           AABB112211|
+-------------------------------------+
-- rtrim
SELECT rtrim('    Feathers   ');
+----------------------+
|rtrim(    Feathers   )|
+----------------------+
|              Feathers|
+----------------------+
-- sentences
SELECT sentences('Hi there! Good morning.');
+-------------------------------------+
|sentences(Hi there! Good morning., , )|
+-------------------------------------+
|                  [[Hi, there], [Go...|
+-------------------------------------+
-- soundex
SELECT soundex('Miller');
+---------------+
|soundex(Miller)|
+---------------+
|           M460|
+---------------+
-- space
SELECT concat(space(2), '1');
+-------------------+
|concat(space(2), 1)|
+-------------------+
|                  1|
+-------------------+
-- split
SELECT split('oneAtwoBthreeC', '[ABC]');
+--------------------------------+
```

```
|split(oneAtwoBthreeC, [ABC], -1)|
+--------------------------------+
|             [one, two, three, ]|
+--------------------------------+
SELECT split('oneAtwoBthreeC', '[ABC]', -1);
+--------------------------------+
|split(oneAtwoBthreeC, [ABC], -1)|
+--------------------------------+
|             [one, two, three, ]|
+--------------------------------+
SELECT split('oneAtwoBthreeC', '[ABC]', 2);
+--------------------------------+
|split(oneAtwoBthreeC, [ABC], 2)|
+--------------------------------+
|              [one, twoBthreeC]|
+--------------------------------+
-- split_part
SELECT split_part('11.12.13', '.', 3);
+---------------------------+
|split_part(11.12.13, ., 3)|
+---------------------------+
|                         13|
+---------------------------+
-- startswith
SELECT startswith('Feathers', 'Fea');
+----------------------------+
|startswith(Feathers, Fea)|
+----------------------------+
|                        true|
+----------------------------+
SELECT startswith('Feathers', 'SQL');
+--------------------------+
|startswith(Feathers, SQL)|
+--------------------------+
|                     false|
+--------------------------+
SELECT startswith('Feathers', null);
+--------------------------+
|startswith(Feathers, NULL)|
+--------------------------+
|                      NULL|
+--------------------------+
SELECT startswith(x'537061726b2053514c', x'537061726b');
+------------------------------------------------+
```

```
|startswith(X'537061726B2053514C', X'537061726B')|
+------------------------------------------------+
|                                            true|
+------------------------------------------------+
SELECT startswith(x'537061726b2053514c', x'53514c');
+---------------------------------------+
|startswith(X'537061726B2053514C', X'53514C')|
+---------------------------------------+
|                                  false|
+---------------------------------------+
-- substr
SELECT substr('Feathers', 5);
+------------------------------+
|substr(Feathers, 5, 2147483647)|
+------------------------------+
|                         hers |
+------------------------------+
SELECT substr('Feathers', -3);
+-------------------------------+
|substr(Feathers, -3, 2147483647)|
+-------------------------------+
|                            ers|
+-------------------------------+
SELECT substr('Feathers', 5, 1);
+----------------------+
|substr(Feathers, 5, 1)|
+----------------------+
|                     h|
+----------------------+
SELECT substr('Feathers' FROM 5);
+---------------------------------+
|substring(Feathers, 5, 2147483647)|
+---------------------------------+
|                            hers |
+---------------------------------+
SELECT substr('Feathers' FROM -3);
+----------------------------------+
|substring(Feathers, -3, 2147483647)|
+----------------------------------+
|                               ers|
+----------------------------------+
SELECT substr('Feathers' FROM 5 FOR 1);
+-------------------------+
|substring(Feathers, 5, 1)|
```

```
+-------------------------+
|                        h|
+-------------------------+
-- substring
SELECT substring('Feathers', 5);
+---------------------------------+
|substring(Feathers, 5, 2147483647)|
+---------------------------------+
|                             hers |
+---------------------------------+
SELECT substring('Feathers', -3);
+----------------------------------+
|substring(Feathers, -3, 2147483647)|
+----------------------------------+
|                                ers|
+----------------------------------+
SELECT substring('Feathers', 5, 1);
+-------------------------+
|substring(Feathers, 5, 1)|
+-------------------------+
|                        h|
+-------------------------+
SELECT substring('Feathers' FROM 5);
+---------------------------------+
|substring(Feathers, 5, 2147483647)|
+---------------------------------+
|                             hers |
+---------------------------------+
SELECT substring('Feathers' FROM -3);
+----------------------------------+
|substring(Feathers, -3, 2147483647)|
+----------------------------------+
|                                ers|
+----------------------------------+
SELECT substring('Feathers' FROM 5 FOR 1);
+-------------------------+
|substring(Feathers, 5, 1)|
+-------------------------+
|                        h|
+-------------------------+
-- substring_index
SELECT substring_index('www.apache.org', '.', 2);
+-----------------------------------+
|substring_index(www.apache.org, ., 2)|
```

```
+------------------------------------+
|                           www.apache|
+------------------------------------+
-- to_binary
SELECT to_binary('abc', 'utf-8');
+---------------------+
|to_binary(abc, utf-8)|
+---------------------+
|           [61 62 63]|
+---------------------+
-- to_char
SELECT to_char(454, '999');
+-----------------+
|to_char(454, 999)|
+-----------------+
|              454|
+-----------------+
SELECT to_char(454.00, '000D00');
+------------------------+
|to_char(454.00, 000D00)|
+------------------------+
|                 454.00|
+------------------------+
SELECT to_char(12454, '99G999');
+----------------------+
|to_char(12454, 99G999)|
+----------------------+
|                12,454|
+----------------------+
SELECT to_char(78.12, '$99.99');
+----------------------+
|to_char(78.12, $99.99)|
+----------------------+
|                $78.12|
+----------------------+
SELECT to_char(-12454.8, '99G999D9S');
+----------------------------+
|to_char(-12454.8, 99G999D9S)|
+----------------------------+
|                   12,454.8-|
+----------------------------+
-- to_number
SELECT to_number('454', '999');
+------------------+
```

```
|to_number(454, 999)|
+-------------------+
|                454|
+-------------------+
SELECT to_number('454.00', '000.00');
+-------------------------+
|to_number(454.00, 000.00)|
+-------------------------+
|                   454.00|
+-------------------------+
SELECT to_number('12,454', '99,999');
+-------------------------+
|to_number(12,454, 99,999)|
+-------------------------+
|                    12454|
+-------------------------+
SELECT to_number('$78.12', '$99.99');
+-------------------------+
|to_number($78.12, $99.99)|
+-------------------------+
|                    78.12|
+-------------------------+
SELECT to_number('12,454.8-', '99,999.9S');
+-------------------------------+
|to_number(12,454.8-, 99,999.9S)|
+-------------------------------+
|                       -12454.8|
+-------------------------------+
-- to_varchar
SELECT to_varchar(454, '999');
+-----------------+
|to_char(454, 999)|
+-----------------+
|              454|
+-----------------+
SELECT to_varchar(454.00, '000D00');
+-----------------------+
|to_char(454.00, 000D00)|
+-----------------------+
|                 454.00|
+-----------------------+
SELECT to_varchar(12454, '99G999');
+----------------------+
|to_char(12454, 99G999)|
```

```
+----------------------+
|               12,454|
+----------------------+
SELECT to_varchar(78.12, '$99.99');
+----------------------+
|to_char(78.12, $99.99)|
+----------------------+
|               $78.12|
+----------------------+
SELECT to_varchar(-12454.8, '99G999D9S');
+---------------------------+
|to_char(-12454.8, 99G999D9S)|
+---------------------------+
|                  12,454.8-|
+---------------------------+
-- translate
SELECT translate('AaBbCc', 'abc', '123');
+---------------------------+
|translate(AaBbCc, abc, 123)|
+---------------------------+
|                     A1B2C3|
+---------------------------+
-- try_to_binary
SELECT try_to_binary('abc', 'utf-8');
+-------------------------+
|try_to_binary(abc, utf-8)|
+-------------------------+
|              [61 62 63]|
+-------------------------+
select try_to_binary('a!', 'base64');
+-------------------------+
|try_to_binary(a!, base64)|
+-------------------------+
|                     NULL|
+-------------------------+
select try_to_binary('abc', 'invalidFormat');
+---------------------------------+
|try_to_binary(abc, invalidFormat)|
+---------------------------------+
|                             NULL|
+---------------------------------+
-- try_to_number
SELECT try_to_number('454', '999');
+----------------------+
```

```
|try_to_number(454, 999)|
+-----------------------+
|                    454|
+-----------------------+
SELECT try_to_number('454.00', '000.00');
+----------------------------+
|try_to_number(454.00, 000.00)|
+----------------------------+
|                      454.00|
+----------------------------+
SELECT try_to_number('12,454', '99,999');
+----------------------------+
|try_to_number(12,454, 99,999)|
+----------------------------+
|                       12454|
+----------------------------+
SELECT try_to_number('$78.12', '$99.99');
+----------------------------+
|try_to_number($78.12, $99.99)|
+----------------------------+
|                       78.12|
+----------------------------+
SELECT try_to_number('12,454.8-', '99,999.9S');
+-----------------------------------+
|try_to_number(12,454.8-, 99,999.9S)|
+-----------------------------------+
|                          -12454.8|
+-----------------------------------+
-- ucase
SELECT ucase('Feathers');
+---------------+
|ucase(Feathers)|
+---------------+
|       FEATHERS|
+---------------+
-- unbase64
SELECT unbase64('U3BhcmsgU1FM');
+----------------------+
|unbase64(U3BhcmsgU1FM)|
+----------------------+
|  [53 70 61 72 6B 2...|
+----------------------+
-- upper
SELECT upper('Feathers');
```

```
+---------------+
|upper(Feathers)|
+---------------+
|       FEATHERS|
+---------------+
```

Funciones de fecha y hora

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando SQL, consulte the section called “Comandos SQL compatibles”.

Función	Descripción
add_months(start_date, num_months)	Devuelve la fecha que es `num_months` después de `start_date` .
convert_timezone([sourceTz,]targetTz, sourceTs)	Convierte la marca de tiempo `sourceTs` sin zona horaria `sourceTz` de a la zona horaria `targetTz`.
curdate()	Devuelve la fecha actual al inicio de la evaluación de la consulta. Todas las llamadas a curdate dentro de la misma consulta devuelven el mismo valor.
current_date()	Devuelve la fecha actual al inicio de la evaluación de la consulta. Todas las llamadas a current_date dentro de la misma consulta devuelven el mismo valor.
current_date	Devuelve la fecha actual al inicio de la evaluación de la consulta.
current_timestamp()	Devuelve la marca de tiempo actual al inicio de la evaluación de la consulta. Todas las

Función	Descripción
	llamadas a current_timestamp dentro de la misma consulta devuelven el mismo valor.
current_timestamp	Devuelve la marca de tiempo actual al inicio de la evaluación de la consulta.
current_timezone()	Devuelve la zona horaria local de la sesión actual.
date_add(start_date, num_days)	Devuelve la fecha que es `num_days` después de `start_date` .
date_diff(endDate, startDate)	Devuelve el número de días comprendido entre `startDate` y `endDate`.
date_format(timestamp, fmt)	Se convierte `timestamp` en un valor de cadena en el formato especificado por el formato de fecha `fmt`.
date_from_unix_date(days)	Crea una fecha a partir del número de días transcurridos desde 1970-01-01.
date_part(field, source)	Extrae una parte de la fecha/marca de tiempo u origen del intervalo.
date_sub(start_date, num_days)	Devuelve la fecha que es `num_days` antes de `start_date` .
date_trunc(fmt, ts)	Devuelve la marca de tiempo `ts` truncada a la unidad especificada por el modelo de formato `fmt`.
dateadd(start_date, num_days)	Devuelve la fecha que es `num_days` después de `start_date` .
datediff(endDate, startDate)	Devuelve el número de días comprendido entre `startDate` y `endDate`.

Función	Descripción
datepart(field, source)	Extrae una parte de la fecha/marca de tiempo u origen del intervalo.
day(date)	Devuelve el día del mes de la fecha/marca horaria.
dayofmonth(date)	Devuelve el día del mes de la fecha/marca horaria.
dayofweek(date)	Devuelve el día de la semana correspondiente a la fecha y hora (1 = domingo, 2 = lunes, ..., 7 = sábado).
dayofyear(date)	Devuelve el día del año de la fecha/marca horaria.
extract(field FROM source)	Extrae una parte de la fecha/marca de tiempo u origen del intervalo.
from_unixtime(unix_time[, fmt])	Devuelve el valor `unix_time` especificado en `fmt`.
from_utc_timestamp(timestamp, timezone)	Dada una marca de tiempo como “2017-07-14 02:40:00.0”, la interpreta como una hora en UTC y representa esa hora como una marca de tiempo en la zona horaria determinada. Por ejemplo, “GMT+1” daría como resultado “2017-07-14 03:40:00.0”.
hour(timestamp)	Devuelve el componente horario de la cadena/marca de tiempo.
last_day(date)	Devuelve el último día del mes al que pertenece la fecha.

Función	Descripción
localtimestamp()	Devuelve la marca de tiempo actual sin zona horaria al inicio de la evaluación de la consulta. Todas las llamadas a localtimestamp dentro de la misma consulta devuelven el mismo valor.
localtimestamp	Devuelve la fecha y hora local actual en la zona horaria de la sesión al inicio de la evaluación de la consulta.
make_date(year, month, day)	Crea una fecha a partir de los campos de año, mes y día.
make_dt_interval([days[, hours[, mins[, secs]]]])	Hace que DayTimeIntervalType dure en días, horas, minutos y segundos.
make_interval([years[, months[, weeks[, days[, hours[, mins[, secs]]]]]]])	Hace intervalos entre años, meses, semanas, días, horas, minutos y segundos.
make_timestamp(year, month, day, hour, min, sec[, timezone])	Crea una marca de tiempo a partir de los campos de año, mes, día, hora, minuto, segundo y zona horaria.
make_timestamp_ltz(year, month, day, hour, min, sec[, timezone])	Crea la marca de tiempo actual con la zona horaria local a partir de los campos de año, mes, día, hora, minutos, segundos y zona horaria.
make_timestamp_ntz(year, month, day, hour, min, sec)	Crea una fecha y hora local a partir de los campos de año, mes, día, hora, minutos y segundos.
make_ym_interval([years[, months]])	Hace un intervalo año-mes a partir de años, meses.
minute(timestamp)	Devuelve el componente de minutos de la cadena/marca de tiempo.

Función	Descripción
month(date)	Devuelve el componente mensual de la fecha/marca horaria.
months_between(timestamp1, timestamp2[, roundOff])	Si `timestamp1` es posterior a `timestamp2`, entonces el resultado es positivo. Si `timestamp1` y `timestamp2` son el mismo día del mes, o ambos son el último día del mes, se ignorará la hora del día. De lo contrario, la diferencia se calcula en función de 31 días al mes y se redondea a 8 dígitos, a menos que exista roundOff=false.
next_day(start_date, day_of_week)	Devuelve la primera fecha posterior a `start_date` y cuyo nombre es el indicado. La función devuelve NULL si al menos uno de los parámetros de entrada es NULL.
now()	Devuelve la marca de tiempo actual al inicio de la evaluación de la consulta.
quarter(date)	Devuelve el trimestre del año correspondiente a la fecha, en un intervalo de 1 a 4.
second(timestamp)	Devuelve el segundo componente de la cadena/marca de tiempo.
session_window(time_column, gap_duration)	Genera una ventana de sesión con una marca de tiempo que especifica la duración de la columna y el intervalo. Consulte “Tipos de intervalos” en el documento de la guía de transmisión estructurada para obtener una explicación detallada y ejemplos.
timestamp_micros(microseconds)	Crea una marca de tiempo a partir del número de microsegundos transcurridos desde la época UTC.

Función	Descripción
timestamp_millis(milliseconds)	Crea una marca de tiempo a partir del número de milisegundos transcurridos desde la época UTC.
timestamp_seconds(seconds)	Crea una marca de tiempo a partir del número de segundos (puede ser fraccionario) desde la época UTC.
to_date(date_str[, fmt])	Analiza la expresión `date_str` con la expresión `fmt` hasta una fecha. Devuelve un valor nulo con una entrada no válida. De forma predeterminada, sigue las reglas de selección hasta una fecha si `fmt` se omite.
to_timestamp(timestamp_str[, fmt])	Analiza la expresión `timestamp_str` junto con la expresión `fmt` en una marca de tiempo. Devuelve un valor nulo con una entrada no válida. De forma predeterminada, sigue las reglas de conversión a una marca de tiempo si `fmt` se omite.
to_timestamp_ltz(timestamp_str[, fmt])	Analiza la expresión `timestamp_str` con la expresión `fmt` para convertirla en una marca de tiempo con la zona horaria local. Devuelve un valor nulo con una entrada no válida. De forma predeterminada, sigue las reglas de conversión a una marca de tiempo si `fmt` se omite.
to_timestamp_ntz(timestamp_str[, fmt])	Analiza la expresión `timestamp_str` con la expresión `fmt` para convertirla en una marca de tiempo sin zona horaria. Devuelve un valor nulo con una entrada no válida. De forma predeterminada, sigue las reglas de conversió n a una marca de tiempo si `fmt` se omite.

Función	Descripción
to_unix_timestamp(timeExp[, fmt])	Devuelve la marca de tiempo de UNIX de la hora dada.
to_utc_timestamp(timestamp, timezone)	Dada una marca de tiempo como “2017-07-14 02:40:00.0”, la interpreta como una hora de la zona horaria determinada y representa esa hora como una marca de tiempo en UTC. Por ejemplo, “GMT+1” daría como resultado “2017-07-14 01:40:00.0”.
trunc(date, fmt)	Devuelve `date` con la parte horaria del día truncada a la unidad especificada por el modelo de formato `fmt`.
try_to_timestamp(timestamp_str[, fmt])	Analiza la expresión `timestamp_str` junto con la expresión `fmt` en una marca de tiempo.
unix_date(date)	Devuelve el número de días transcurridos desde 1970-01-01.
unix_micros(timestamp)	Devuelve el número de microsegundos transcurridos desde 1970-01-01 00:00:00 UTC.
unix_millis(timestamp)	Devuelve el número de milisegundos transcurridos desde 1970-01-01 00:00:00 UTC. Trunca los niveles de precisión más altos.
unix_seconds(timestamp)	Devuelve el número de segundos transcurridos desde 1970-01-01 00:00:00 UTC. Trunca los niveles de precisión más altos.
unix_timestamp([timeExp[, fmt]])	Devuelve la marca de tiempo de UNIX de la hora actual o especificada.
weekday(date)	Devuelve el día de la semana correspondiente a la fecha y hora (0 = lunes, 1 = martes, ..., 6 = domingo).

Función	Descripción
weekofyear(date)	Devuelve la semana del año de la fecha indicada. Se considera que una semana comienza un lunes y la semana 1 es la primera semana con más de 3 días.
window(time_column, window_duration[, slide_duration[, start_time]])	Clasifica las filas en una o más ventanas de tiempo con una columna que especifique la marca de tiempo. Los inicios de las ventanas son inclusivos, pero los finales son exclusivos, por ejemplo, las 12:05 estarán en la ventana [12:05,12:10) pero no en [12:00,12:05). Windows admite una precisión de microsegundos. No se admite Windows para el orden de los meses. Consulte “Operaciones de intervalo en tiempos de eventos” en el documento de la guía de transmisión estructurada para obtener una explicación detallada y ejemplos.
window_time(window_column)	Extrae el valor de tiempo de la columna de la ventana de tiempo/sesión que se puede utilizar como valor de tiempo del evento de la ventana. El tiempo extraído es (window.end - 1), lo que refleja el hecho de que los intervalos de agregación tienen un límite superior exclusivo (inicio, final). Consulte “Operaciones de intervalo en tiempos de eventos” en el documento de la guía de transmisión estructurada para obtener una explicación detallada y ejemplos.
year(date)	Devuelve el componente anual de la fecha/marca horaria.

Ejemplos

```
-- add_months
SELECT add_months('2016-08-31', 1);
+-------------------------+
|add_months(2016-08-31, 1)|
+-------------------------+
|               2016-09-30|
+-------------------------+
-- convert_timezone
SELECT convert_timezone('Europe/Brussels', 'America/Los_Angeles',
 timestamp_ntz'2021-12-06 00:00:00');
+-----------------------------------------------------------------------------------------------
+
|convert_timezone(Europe/Brussels, America/Los_Angeles, TIMESTAMP_NTZ '2021-12-06
 00:00:00')|
+-----------------------------------------------------------------------------------------------
+
|                                                                                2021-12-05
 15:00:00|
+-----------------------------------------------------------------------------------------------
+
SELECT convert_timezone('Europe/Brussels', timestamp_ntz'2021-12-05 15:00:00');
+-----------------------------------------------------------------------------------------------
+
|convert_timezone(current_timezone(), Europe/Brussels, TIMESTAMP_NTZ '2021-12-05
 15:00:00')|
+-----------------------------------------------------------------------------------------------
+
|                                                                              2021-12-05
 07:00:00|
+-----------------------------------------------------------------------------------------------
+
-- curdate
SELECT curdate();
+--------------+
|current_date()|
+--------------+
|    2024-02-24|
+--------------+
-- current_date
SELECT current_date();
+--------------+
|current_date()|
+--------------+
```

```
|    2024-02-24|
+--------------+
SELECT current_date;
+--------------+
|current_date()|
+--------------+
|    2024-02-24|
+--------------+
-- current_timestamp
SELECT current_timestamp();
+--------------------+
| current_timestamp()|
+--------------------+
|2024-02-24 16:36:...|
+--------------------+
SELECT current_timestamp;
+--------------------+
| current_timestamp()|
+--------------------+
|2024-02-24 16:36:...|
+--------------------+
-- current_timezone
SELECT current_timezone();
+------------------+
|current_timezone()|
+------------------+
|        Asia/Seoul|
+------------------+
-- date_add
SELECT date_add('2016-07-30', 1);
+-----------------------+
|date_add(2016-07-30, 1)|
+-----------------------+
|             2016-07-31|
+-----------------------+
-- date_diff
SELECT date_diff('2009-07-31', '2009-07-30');
+---------------------------------+
|date_diff(2009-07-31, 2009-07-30)|
+---------------------------------+
|                                1|
+---------------------------------+
SELECT date_diff('2009-07-30', '2009-07-31');
+---------------------------------+
```

```
|date_diff(2009-07-30, 2009-07-31)|
+---------------------------------+
|                               -1|
+---------------------------------+
-- date_format
SELECT date_format('2016-04-08', 'y');
+--------------------------+
|date_format(2016-04-08, y)|
+--------------------------+
|                      2016|
+--------------------------+
-- date_from_unix_date
SELECT date_from_unix_date(1);
+----------------------+
|date_from_unix_date(1)|
+----------------------+
|            1970-01-02|
+----------------------+
-- date_part
SELECT date_part('YEAR', TIMESTAMP '2019-08-12 01:00:00.123456');
+-----------------------------------------------------------+
|date_part(YEAR, TIMESTAMP '2019-08-12 01:00:00.123456')|
+-----------------------------------------------------------+
|                                                       2019|
+-----------------------------------------------------------+
SELECT date_part('week', timestamp'2019-08-12 01:00:00.123456');
+-----------------------------------------------------------+
|date_part(week, TIMESTAMP '2019-08-12 01:00:00.123456')|
+-----------------------------------------------------------+
|                                                         33|
+-----------------------------------------------------------+
SELECT date_part('doy', DATE'2019-08-12');
+---------------------------------+
|date_part(doy, DATE '2019-08-12')|
+---------------------------------+
|                              224|
+---------------------------------+
SELECT date_part('SECONDS', timestamp'2019-10-01 00:00:01.000001');
+-------------------------------------------------------------+
|date_part(SECONDS, TIMESTAMP '2019-10-01 00:00:01.000001')|
+-------------------------------------------------------------+
|                                                     1.000001|
+-------------------------------------------------------------+
SELECT date_part('days', interval 5 days 3 hours 7 minutes);
```

```
+-----------------------------------------------+
|date_part(days, INTERVAL '5 03:07' DAY TO MINUTE)|
+-----------------------------------------------+
|                                              5|
+-----------------------------------------------+
SELECT date_part('seconds', interval 5 hours 30 seconds 1 milliseconds 1 microseconds);
+-------------------------------------------------------------+
|date_part(seconds, INTERVAL '05:00:30.001001' HOUR TO SECOND)|
+-------------------------------------------------------------+
|                                                    30.001001|
+-------------------------------------------------------------+
SELECT date_part('MONTH', INTERVAL '2021-11' YEAR TO MONTH);
+-----------------------------------------------+
|date_part(MONTH, INTERVAL '2021-11' YEAR TO MONTH)|
+-----------------------------------------------+
|                                             11|
+-----------------------------------------------+
SELECT date_part('MINUTE', INTERVAL '123 23:55:59.002001' DAY TO SECOND);
+-------------------------------------------------------------+
|date_part(MINUTE, INTERVAL '123 23:55:59.002001' DAY TO SECOND)|
+-------------------------------------------------------------+
|                                                           55|
+-------------------------------------------------------------+
-- date_sub
SELECT date_sub('2016-07-30', 1);
+-----------------------+
|date_sub(2016-07-30, 1)|
+-----------------------+
|             2016-07-29|
+-----------------------+
-- date_trunc
SELECT date_trunc('YEAR', '2015-03-05T09:32:05.359');
+----------------------------------------+
|date_trunc(YEAR, 2015-03-05T09:32:05.359)|
+----------------------------------------+
|                     2015-01-01 00:00:00|
+----------------------------------------+
SELECT date_trunc('MM', '2015-03-05T09:32:05.359');
+-------------------------------------+
|date_trunc(MM, 2015-03-05T09:32:05.359)|
+-------------------------------------+
|                    2015-03-01 00:00:00|
+-------------------------------------+
SELECT date_trunc('DD', '2015-03-05T09:32:05.359');
```

```
+--------------------------------------+
|date_trunc(DD, 2015-03-05T09:32:05.359)|
+--------------------------------------+
|                   2015-03-05 00:00:00|
+--------------------------------------+
SELECT date_trunc('HOUR', '2015-03-05T09:32:05.359');
+---------------------------------------+
|date_trunc(HOUR, 2015-03-05T09:32:05.359)|
+---------------------------------------+
|                     2015-03-05 09:00:00|
+---------------------------------------+
SELECT date_trunc('MILLISECOND', '2015-03-05T09:32:05.123456');
+-----------------------------------------------------+
|date_trunc(MILLISECOND, 2015-03-05T09:32:05.123456)|
+-----------------------------------------------------+
|                              2015-03-05 09:32:...|
+-----------------------------------------------------+
-- dateadd
SELECT dateadd('2016-07-30', 1);
+----------------------+
|date_add(2016-07-30, 1)|
+----------------------+
|             2016-07-31|
+----------------------+
-- datediff
SELECT datediff('2009-07-31', '2009-07-30');
+--------------------------------+
|datediff(2009-07-31, 2009-07-30)|
+--------------------------------+
|                               1|
+--------------------------------+
SELECT datediff('2009-07-30', '2009-07-31');
+--------------------------------+
|datediff(2009-07-30, 2009-07-31)|
+--------------------------------+
|                              -1|
+--------------------------------+
-- datepart
SELECT datepart('YEAR', TIMESTAMP '2019-08-12 01:00:00.123456');
+-------------------------------------------------------------+
|datepart(YEAR FROM TIMESTAMP '2019-08-12 01:00:00.123456')|
+-------------------------------------------------------------+
|                                                         2019|
+-------------------------------------------------------------+
```

```
SELECT datepart('week', timestamp'2019-08-12 01:00:00.123456');
+-----------------------------------------------------------+
|datepart(week FROM TIMESTAMP '2019-08-12 01:00:00.123456')|
+-----------------------------------------------------------+
|                                                         33|
+-----------------------------------------------------------+
SELECT datepart('doy', DATE'2019-08-12');
+-----------------------------------+
|datepart(doy FROM DATE '2019-08-12')|
+-----------------------------------+
|                                224|
+-----------------------------------+
SELECT datepart('SECONDS', timestamp'2019-10-01 00:00:01.000001');
+-------------------------------------------------------------+
|datepart(SECONDS FROM TIMESTAMP '2019-10-01 00:00:01.000001')|
+-------------------------------------------------------------+
|                                                     1.000001|
+-------------------------------------------------------------+
SELECT datepart('days', interval 5 days 3 hours 7 minutes);
+---------------------------------------------------+
|datepart(days FROM INTERVAL '5 03:07' DAY TO MINUTE)|
+---------------------------------------------------+
|                                                  5|
+---------------------------------------------------+
SELECT datepart('seconds', interval 5 hours 30 seconds 1 milliseconds 1 microseconds);
+-----------------------------------------------------------------+
|datepart(seconds FROM INTERVAL '05:00:30.001001' HOUR TO SECOND)|
+-----------------------------------------------------------------+
|                                                        30.001001|
+-----------------------------------------------------------------+
SELECT datepart('MONTH', INTERVAL '2021-11' YEAR TO MONTH);
+---------------------------------------------------+
|datepart(MONTH FROM INTERVAL '2021-11' YEAR TO MONTH)|
+---------------------------------------------------+
|                                                 11|
+---------------------------------------------------+
SELECT datepart('MINUTE', INTERVAL '123 23:55:59.002001' DAY TO SECOND);
+-------------------------------------------------------------------+
|datepart(MINUTE FROM INTERVAL '123 23:55:59.002001' DAY TO SECOND)|
+-------------------------------------------------------------------+
|                                                                 55|
+-------------------------------------------------------------------+
-- day
SELECT day('2009-07-30');
```

```
+---------------+
|day(2009-07-30)|
+---------------+
|             30|
+---------------+
-- dayofmonth
SELECT dayofmonth('2009-07-30');
+----------------------+
|dayofmonth(2009-07-30)|
+----------------------+
|                    30|
+----------------------+
-- dayofweek
SELECT dayofweek('2009-07-30');
+---------------------+
|dayofweek(2009-07-30)|
+---------------------+
|                    5|
+---------------------+
-- dayofyear
SELECT dayofyear('2016-04-09');
+---------------------+
|dayofyear(2016-04-09)|
+---------------------+
|                  100|
+---------------------+
-- extract
SELECT extract(YEAR FROM TIMESTAMP '2019-08-12 01:00:00.123456');
+-----------------------------------------------------------+
|extract(YEAR FROM TIMESTAMP '2019-08-12 01:00:00.123456')|
+-----------------------------------------------------------+
|                                                       2019|
+-----------------------------------------------------------+
SELECT extract(week FROM timestamp'2019-08-12 01:00:00.123456');
+-----------------------------------------------------------+
|extract(week FROM TIMESTAMP '2019-08-12 01:00:00.123456')|
+-----------------------------------------------------------+
|                                                         33|
+-----------------------------------------------------------+
SELECT extract(doy FROM DATE'2019-08-12');
+----------------------------------+
|extract(doy FROM DATE '2019-08-12')|
+----------------------------------+
|                               224|
```

```
+----------------------------------+
SELECT extract(SECONDS FROM timestamp'2019-10-01 00:00:01.000001');
+-------------------------------------------------------------+
|extract(SECONDS FROM TIMESTAMP '2019-10-01 00:00:01.000001')|
+-------------------------------------------------------------+
|                                                     1.000001|
+-------------------------------------------------------------+
SELECT extract(days FROM interval 5 days 3 hours 7 minutes);
+---------------------------------------------------+
|extract(days FROM INTERVAL '5 03:07' DAY TO MINUTE)|
+---------------------------------------------------+
|                                                  5|
+---------------------------------------------------+
SELECT extract(seconds FROM interval 5 hours 30 seconds 1 milliseconds 1 microseconds);
+-----------------------------------------------------------------+
|extract(seconds FROM INTERVAL '05:00:30.001001' HOUR TO SECOND)|
+-----------------------------------------------------------------+
|                                                       30.001001|
+-----------------------------------------------------------------+
SELECT extract(MONTH FROM INTERVAL '2021-11' YEAR TO MONTH);
+---------------------------------------------------+
|extract(MONTH FROM INTERVAL '2021-11' YEAR TO MONTH)|
+---------------------------------------------------+
|                                                 11|
+---------------------------------------------------+
SELECT extract(MINUTE FROM INTERVAL '123 23:55:59.002001' DAY TO SECOND);
+-------------------------------------------------------------------+
|extract(MINUTE FROM INTERVAL '123 23:55:59.002001' DAY TO SECOND)|
+-------------------------------------------------------------------+
|                                                                 55|
+-------------------------------------------------------------------+
-- from_unixtime
SELECT from_unixtime(0, 'yyyy-MM-dd HH:mm:ss');
+-------------------------------------+
|from_unixtime(0, yyyy-MM-dd HH:mm:ss)|
+-------------------------------------+
|                  1970-01-01 09:00:00|
+-------------------------------------+
SELECT from_unixtime(0);
+-------------------------------------+
|from_unixtime(0, yyyy-MM-dd HH:mm:ss)|
+-------------------------------------+
|                  1970-01-01 09:00:00|
+-------------------------------------+
```

```
-- from_utc_timestamp
SELECT from_utc_timestamp('2016-08-31', 'Asia/Seoul');
+-----------------------------------------+
|from_utc_timestamp(2016-08-31, Asia/Seoul)|
+-----------------------------------------+
|                      2016-08-31 09:00:00|
+-----------------------------------------+
-- hour
SELECT hour('2009-07-30 12:58:59');
+-------------------------+
|hour(2009-07-30 12:58:59)|
+-------------------------+
|                       12|
+-------------------------+
-- last_day
SELECT last_day('2009-01-12');
+--------------------+
|last_day(2009-01-12)|
+--------------------+
|          2009-01-31|
+--------------------+
-- localtimestamp
SELECT localtimestamp();
+--------------------+
|    localtimestamp()|
+--------------------+
|2024-02-24 16:36:...|
+--------------------+
-- make_date
SELECT make_date(2013, 7, 15);
+----------------------+
|make_date(2013, 7, 15)|
+----------------------+
|            2013-07-15|
+----------------------+
SELECT make_date(2019, 7, NULL);
+------------------------+
|make_date(2019, 7, NULL)|
+------------------------+
|                    NULL|
+------------------------+
-- make_dt_interval
SELECT make_dt_interval(1, 12, 30, 01.001001);
+-------------------------------------+
```

```
|make_dt_interval(1, 12, 30, 1.001001)|
+-------------------------------------+
|                 INTERVAL '1 12:30...|
+-------------------------------------+
SELECT make_dt_interval(2);
+-----------------------------------+
|make_dt_interval(2, 0, 0, 0.000000)|
+-----------------------------------+
|               INTERVAL '2 00:00...|
+-----------------------------------+
SELECT make_dt_interval(100, null, 3);
+---------------------------------------+
|make_dt_interval(100, NULL, 3, 0.000000)|
+---------------------------------------+
|                                   NULL|
+---------------------------------------+
-- make_interval
SELECT make_interval(100, 11, 1, 1, 12, 30, 01.001001);
+-----------------------------------------------+
|make_interval(100, 11, 1, 1, 12, 30, 1.001001)|
+-----------------------------------------------+
|                          100 years 11 mont...|
+-----------------------------------------------+
SELECT make_interval(100, null, 3);
+-----------------------------------------------+
|make_interval(100, NULL, 3, 0, 0, 0, 0.000000)|
+-----------------------------------------------+
|                                           NULL|
+-----------------------------------------------+
SELECT make_interval(0, 1, 0, 1, 0, 0, 100.000001);
+-------------------------------------------+
|make_interval(0, 1, 0, 1, 0, 0, 100.000001)|
+-------------------------------------------+
|                       1 months 1 days 1...|
+-------------------------------------------+
-- make_timestamp
SELECT make_timestamp(2014, 12, 28, 6, 30, 45.887);
+-----------------------------------------+
|make_timestamp(2014, 12, 28, 6, 30, 45.887)|
+-----------------------------------------+
|                      2014-12-28 06:30:...|
+-----------------------------------------+
SELECT make_timestamp(2014, 12, 28, 6, 30, 45.887, 'CET');
+-------------------------------------------------+
```

```
|make_timestamp(2014, 12, 28, 6, 30, 45.887, CET)|
+------------------------------------------------+
|                           2014-12-28 14:30:...|
+------------------------------------------------+
SELECT make_timestamp(2019, 6, 30, 23, 59, 60);
+--------------------------------------+
|make_timestamp(2019, 6, 30, 23, 59, 60)|
+--------------------------------------+
|                    2019-07-01 00:00:00|
+--------------------------------------+
SELECT make_timestamp(2019, 6, 30, 23, 59, 1);
+--------------------------------------+
|make_timestamp(2019, 6, 30, 23, 59, 1)|
+--------------------------------------+
|                   2019-06-30 23:59:01|
+--------------------------------------+
SELECT make_timestamp(null, 7, 22, 15, 30, 0);
+--------------------------------------+
|make_timestamp(NULL, 7, 22, 15, 30, 0)|
+--------------------------------------+
|                                  NULL|
+--------------------------------------+
-- make_timestamp_ltz
SELECT make_timestamp_ltz(2014, 12, 28, 6, 30, 45.887);
+-----------------------------------------------+
|make_timestamp_ltz(2014, 12, 28, 6, 30, 45.887)|
+-----------------------------------------------+
|                          2014-12-28 06:30:...|
+-----------------------------------------------+
SELECT make_timestamp_ltz(2014, 12, 28, 6, 30, 45.887, 'CET');
+-------------------------------------------------------+
|make_timestamp_ltz(2014, 12, 28, 6, 30, 45.887, CET)|
+-------------------------------------------------------+
|                               2014-12-28 14:30:...|
+-------------------------------------------------------+
SELECT make_timestamp_ltz(2019, 6, 30, 23, 59, 60);
+-----------------------------------------+
|make_timestamp_ltz(2019, 6, 30, 23, 59, 60)|
+-----------------------------------------+
|                      2019-07-01 00:00:00|
+-----------------------------------------+
SELECT make_timestamp_ltz(null, 7, 22, 15, 30, 0);
+-----------------------------------------+
|make_timestamp_ltz(NULL, 7, 22, 15, 30, 0)|
```

```
+----------------------------------------+
|                                    NULL|
+----------------------------------------+
-- make_timestamp_ntz
SELECT make_timestamp_ntz(2014, 12, 28, 6, 30, 45.887);
+-----------------------------------------------+
|make_timestamp_ntz(2014, 12, 28, 6, 30, 45.887)|
+-----------------------------------------------+
|                        2014-12-28 06:30:...|
+-----------------------------------------------+
SELECT make_timestamp_ntz(2019, 6, 30, 23, 59, 60);
+----------------------------------------+
|make_timestamp_ntz(2019, 6, 30, 23, 59, 60)|
+----------------------------------------+
|                      2019-07-01 00:00:00|
+----------------------------------------+
SELECT make_timestamp_ntz(null, 7, 22, 15, 30, 0);
+----------------------------------------+
|make_timestamp_ntz(NULL, 7, 22, 15, 30, 0)|
+----------------------------------------+
|                                    NULL|
+----------------------------------------+
-- make_ym_interval
SELECT make_ym_interval(1, 2);
+---------------------+
|make_ym_interval(1, 2)|
+---------------------+
|  INTERVAL '1-2' YE...|
+---------------------+
SELECT make_ym_interval(1, 0);
+---------------------+
|make_ym_interval(1, 0)|
+---------------------+
|  INTERVAL '1-0' YE...|
+---------------------+
SELECT make_ym_interval(-1, 1);
+----------------------+
|make_ym_interval(-1, 1)|
+----------------------+
|   INTERVAL '-0-11' ...|
+----------------------+
SELECT make_ym_interval(2);
+---------------------+
|make_ym_interval(2, 0)|
```

```
+----------------------+
|  INTERVAL '2-0' YE...|
+----------------------+
-- minute
SELECT minute('2009-07-30 12:58:59');
+--------------------------+
|minute(2009-07-30 12:58:59)|
+--------------------------+
|                        58|
+--------------------------+
-- month
SELECT month('2016-07-30');
+-----------------+
|month(2016-07-30)|
+-----------------+
|                7|
+-----------------+
-- months_between
SELECT months_between('1997-02-28 10:30:00', '1996-10-30');
+-----------------------------------------------------+
|months_between(1997-02-28 10:30:00, 1996-10-30, true)|
+-----------------------------------------------------+
|                                           3.94959677|
+-----------------------------------------------------+
SELECT months_between('1997-02-28 10:30:00', '1996-10-30', false);
+------------------------------------------------------+
|months_between(1997-02-28 10:30:00, 1996-10-30, false)|
+------------------------------------------------------+
|                                    3.9495967741935485|
+------------------------------------------------------+
-- next_day
SELECT next_day('2015-01-14', 'TU');
+------------------------+
|next_day(2015-01-14, TU)|
+------------------------+
|              2015-01-20|
+------------------------+
-- now
SELECT now();
+--------------------+
|               now()|
+--------------------+
|2024-02-24 16:36:...|
+--------------------+
```

```
-- quarter
SELECT quarter('2016-08-31');
+-------------------+
|quarter(2016-08-31)|
+-------------------+
|                  3|
+-------------------+
-- second
SELECT second('2009-07-30 12:58:59');
+--------------------------+
|second(2009-07-30 12:58:59)|
+--------------------------+
|                        59|
+--------------------------+
-- session_window
SELECT a, session_window.start, session_window.end, count(*) as cnt FROM VALUES ('A1',
 '2021-01-01 00:00:00'), ('A1', '2021-01-01 00:04:30'), ('A1', '2021-01-01 00:10:00'),
 ('A2', '2021-01-01 00:01:00') AS tab(a, b) GROUP by a, session_window(b, '5 minutes')
 ORDER BY a, start;
+---+-------------------+-------------------+---+
|  a|              start|                end|cnt|
+---+-------------------+-------------------+---+
| A1|2021-01-01 00:00:00|2021-01-01 00:09:30|  2|
| A1|2021-01-01 00:10:00|2021-01-01 00:15:00|  1|
| A2|2021-01-01 00:01:00|2021-01-01 00:06:00|  1|
+---+-------------------+-------------------+---+
SELECT a, session_window.start, session_window.end, count(*) as cnt FROM VALUES ('A1',
 '2021-01-01 00:00:00'), ('A1', '2021-01-01 00:04:30'), ('A1', '2021-01-01 00:10:00'),
 ('A2', '2021-01-01 00:01:00'), ('A2', '2021-01-01 00:04:30') AS tab(a, b) GROUP by a,
 session_window(b, CASE WHEN a = 'A1' THEN '5 minutes' WHEN a = 'A2' THEN '1 minute'
 ELSE '10 minutes' END) ORDER BY a, start;
+---+-------------------+-------------------+---+
|  a|              start|                end|cnt|
+---+-------------------+-------------------+---+
| A1|2021-01-01 00:00:00|2021-01-01 00:09:30|  2|
| A1|2021-01-01 00:10:00|2021-01-01 00:15:00|  1|
| A2|2021-01-01 00:01:00|2021-01-01 00:02:00|  1|
| A2|2021-01-01 00:04:30|2021-01-01 00:05:30|  1|
+---+-------------------+-------------------+---+
-- timestamp_micros
SELECT timestamp_micros(1230219000123123);
+---------------------------------+
|timestamp_micros(1230219000123123)|
+---------------------------------+
```

```
|              2008-12-26 00:30:...|
+----------------------------------+
-- timestamp_millis
SELECT timestamp_millis(1230219000123);
+------------------------------+
|timestamp_millis(1230219000123)|
+------------------------------+
|           2008-12-26 00:30:...|
+------------------------------+
-- timestamp_seconds
SELECT timestamp_seconds(1230219000);
+----------------------------+
|timestamp_seconds(1230219000)|
+----------------------------+
|          2008-12-26 00:30:00|
+----------------------------+
SELECT timestamp_seconds(1230219000.123);
+---------------------------------+
|timestamp_seconds(1230219000.123)|
+---------------------------------+
|             2008-12-26 00:30:...|
+---------------------------------+
-- to_date
SELECT to_date('2009-07-30 04:17:52');
+---------------------------+
|to_date(2009-07-30 04:17:52)|
+---------------------------+
|                 2009-07-30|
+---------------------------+
SELECT to_date('2016-12-31', 'yyyy-MM-dd');
+-------------------------------+
|to_date(2016-12-31, yyyy-MM-dd)|
+-------------------------------+
|                     2016-12-31|
+-------------------------------+
-- to_timestamp
SELECT to_timestamp('2016-12-31 00:12:00');
+---------------------------------+
|to_timestamp(2016-12-31 00:12:00)|
+---------------------------------+
|              2016-12-31 00:12:00|
+---------------------------------+
SELECT to_timestamp('2016-12-31', 'yyyy-MM-dd');
+---------------------------------+
```

```
|to_timestamp(2016-12-31, yyyy-MM-dd)|
+------------------------------------+
|                 2016-12-31 00:00:00|
+------------------------------------+
-- to_timestamp_ltz
SELECT to_timestamp_ltz('2016-12-31 00:12:00');
+-------------------------------------+
|to_timestamp_ltz(2016-12-31 00:12:00)|
+-------------------------------------+
|                  2016-12-31 00:12:00|
+-------------------------------------+
SELECT to_timestamp_ltz('2016-12-31', 'yyyy-MM-dd');
+----------------------------------------+
|to_timestamp_ltz(2016-12-31, yyyy-MM-dd)|
+----------------------------------------+
|                     2016-12-31 00:00:00|
+----------------------------------------+
-- to_timestamp_ntz
SELECT to_timestamp_ntz('2016-12-31 00:12:00');
+-------------------------------------+
|to_timestamp_ntz(2016-12-31 00:12:00)|
+-------------------------------------+
|                  2016-12-31 00:12:00|
+-------------------------------------+
SELECT to_timestamp_ntz('2016-12-31', 'yyyy-MM-dd');
+----------------------------------------+
|to_timestamp_ntz(2016-12-31, yyyy-MM-dd)|
+----------------------------------------+
|                     2016-12-31 00:00:00|
+----------------------------------------+
-- to_unix_timestamp
SELECT to_unix_timestamp('2016-04-08', 'yyyy-MM-dd');
+----------------------------------------+
|to_unix_timestamp(2016-04-08, yyyy-MM-dd)|
+----------------------------------------+
|                                1460041200|
+----------------------------------------+
-- to_utc_timestamp
SELECT to_utc_timestamp('2016-08-31', 'Asia/Seoul');
+----------------------------------------+
|to_utc_timestamp(2016-08-31, Asia/Seoul)|
+----------------------------------------+
|                     2016-08-30 15:00:00|
+----------------------------------------+
```

```
-- trunc
SELECT trunc('2019-08-04', 'week');
+-----------------------+
|trunc(2019-08-04, week)|
+-----------------------+
|             2019-07-29|
+-----------------------+
SELECT trunc('2019-08-04', 'quarter');
+--------------------------+
|trunc(2019-08-04, quarter)|
+--------------------------+
|                2019-07-01|
+--------------------------+
SELECT trunc('2009-02-12', 'MM');
+---------------------+
|trunc(2009-02-12, MM)|
+---------------------+
|           2009-02-01|
+---------------------+
SELECT trunc('2015-10-27', 'YEAR');
+-----------------------+
|trunc(2015-10-27, YEAR)|
+-----------------------+
|             2015-01-01|
+-----------------------+
-- try_to_timestamp
SELECT try_to_timestamp('2016-12-31 00:12:00');
+-------------------------------------+
|try_to_timestamp(2016-12-31 00:12:00)|
+-------------------------------------+
|                  2016-12-31 00:12:00|
+-------------------------------------+
SELECT try_to_timestamp('2016-12-31', 'yyyy-MM-dd');
+---------------------------------------+
|try_to_timestamp(2016-12-31, yyyy-MM-dd)|
+---------------------------------------+
|                     2016-12-31 00:00:00|
+---------------------------------------+
SELECT try_to_timestamp('foo', 'yyyy-MM-dd');
+---------------------------------+
|try_to_timestamp(foo, yyyy-MM-dd)|
+---------------------------------+
|                             NULL|
+---------------------------------+
```

```
-- unix_date
SELECT unix_date(DATE("1970-01-02"));
+---------------------+
|unix_date(1970-01-02)|
+---------------------+
|                    1|
+---------------------+
-- unix_micros
SELECT unix_micros(TIMESTAMP('1970-01-01 00:00:01Z'));
+---------------------------------+
|unix_micros(1970-01-01 00:00:01Z)|
+---------------------------------+
|                          1000000|
+---------------------------------+
-- unix_millis
SELECT unix_millis(TIMESTAMP('1970-01-01 00:00:01Z'));
+---------------------------------+
|unix_millis(1970-01-01 00:00:01Z)|
+---------------------------------+
|                             1000|
+---------------------------------+
-- unix_seconds
SELECT unix_seconds(TIMESTAMP('1970-01-01 00:00:01Z'));
+---------------------------------+
|unix_seconds(1970-01-01 00:00:01Z)|
+---------------------------------+
|                                1|
+---------------------------------+
-- unix_timestamp
SELECT unix_timestamp();
+-----------------------------------------------------------+
|unix_timestamp(current_timestamp(), yyyy-MM-dd HH:mm:ss)|
+-----------------------------------------------------------+
|                                                 1708760216|
+-----------------------------------------------------------+
SELECT unix_timestamp('2016-04-08', 'yyyy-MM-dd');
+-------------------------------------+
|unix_timestamp(2016-04-08, yyyy-MM-dd)|
+-------------------------------------+
|                           1460041200|
+-------------------------------------+
-- weekday
SELECT weekday('2009-07-30');
+------------------+
```

```
|weekday(2009-07-30)|
+-------------------+
|                  3|
+-------------------+
-- weekofyear
SELECT weekofyear('2008-02-20');
+----------------------+
|weekofyear(2008-02-20)|
+----------------------+
|                     8|
+----------------------+
-- window
SELECT a, window.start, window.end, count(*) as cnt FROM VALUES ('A1', '2021-01-01
 00:00:00'), ('A1', '2021-01-01 00:04:30'), ('A1', '2021-01-01 00:06:00'), ('A2',
 '2021-01-01 00:01:00') AS tab(a, b) GROUP by a, window(b, '5 minutes') ORDER BY a,
 start;
+---+-------------------+-------------------+---+
|  a|              start|                end|cnt|
+---+-------------------+-------------------+---+
| A1|2021-01-01 00:00:00|2021-01-01 00:05:00|  2|
| A1|2021-01-01 00:05:00|2021-01-01 00:10:00|  1|
| A2|2021-01-01 00:00:00|2021-01-01 00:05:00|  1|
+---+-------------------+-------------------+---+
SELECT a, window.start, window.end, count(*) as cnt FROM VALUES ('A1', '2021-01-01
 00:00:00'), ('A1', '2021-01-01 00:04:30'), ('A1', '2021-01-01 00:06:00'), ('A2',
 '2021-01-01 00:01:00') AS tab(a, b) GROUP by a, window(b, '10 minutes', '5 minutes')
 ORDER BY a, start;
+---+-------------------+-------------------+---+
|  a|              start|                end|cnt|
+---+-------------------+-------------------+---+
| A1|2020-12-31 23:55:00|2021-01-01 00:05:00|  2|
| A1|2021-01-01 00:00:00|2021-01-01 00:10:00|  3|
| A1|2021-01-01 00:05:00|2021-01-01 00:15:00|  1|
| A2|2020-12-31 23:55:00|2021-01-01 00:05:00|  1|
| A2|2021-01-01 00:00:00|2021-01-01 00:10:00|  1|
+---+-------------------+-------------------+---+
-- window_time
SELECT a, window.start as start, window.end as end, window_time(window), cnt FROM
 (SELECT a, window, count(*) as cnt FROM VALUES ('A1', '2021-01-01 00:00:00'), ('A1',
 '2021-01-01 00:04:30'), ('A1', '2021-01-01 00:06:00'), ('A2', '2021-01-01 00:01:00')
 AS tab(a, b) GROUP by a, window(b, '5 minutes') ORDER BY a, window.start);
+---+-------------------+-------------------+--------------------+---+
|  a|              start|                end| window_time(window)|cnt|
+---+-------------------+-------------------+--------------------+---+
```

```
| A1|2021-01-01 00:00:00|2021-01-01 00:05:00|2021-01-01 00:04:...|  2|
| A1|2021-01-01 00:05:00|2021-01-01 00:10:00|2021-01-01 00:09:...|  1|
| A2|2021-01-01 00:00:00|2021-01-01 00:05:00|2021-01-01 00:04:...|  1|
+---+-------------------+-------------------+-------------------+---+
-- year
SELECT year('2016-07-30');
+----------------+
|year(2016-07-30)|
+----------------+
|            2016|
+----------------+
```

Funciones de agregación

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando SQL, consulte the section called “Comandos SQL compatibles”.

Las funciones agregadas funcionan con valores de filas para realizar cálculos matemáticos como la suma, el promedio, el recuento, los valores mínimos/máximos, la desviación estándar y la estimación, así como algunas operaciones no matemáticas.

Sintaxis

```
aggregate_function(input1 [, input2, ...]) FILTER (WHERE boolean_expression)
```

Parámetros

- `boolean_expression`: especifica cualquier expresión que se evalúe como un resultado de tipo booleano. Se pueden combinar dos o más expresiones mediante los operadores lógicos (AND, OR).

Funciones agregadas de conjuntos ordenados

Estas funciones de agregación utilizan una sintaxis diferente a la de las demás funciones de agregación para especificar una expresión (normalmente el nombre de una columna) con la que ordenar los valores.

Sintaxis

```
{ PERCENTILE_CONT | PERCENTILE_DISC }(percentile) WITHIN GROUP (ORDER BY
 { order_by_expression [ ASC | DESC ] [ NULLS { FIRST | LAST } ] [ , ... ] }) FILTER
 (WHERE boolean_expression)
```

Parámetros

- `percentile`: el percentil del valor que se desea buscar. El percentil debe ser una constante entre 0,0 y 1,0.
- `order_by_expression`: la expresión (normalmente el nombre de una columna) con la que se ordenan los valores antes de agregarlos.
- `boolean_expression`: especifica cualquier expresión que se evalúe como un resultado de tipo booleano. Se pueden combinar dos o más expresiones mediante los operadores lógicos (AND, OR).

Ejemplos

```
CREATE OR REPLACE TEMPORARY VIEW basic_pays AS SELECT * FROM VALUES
('Jane Doe','Accounting',8435),
('Akua Mansa','Accounting',9998),
('John Doe','Accounting',8992),
('Juan Li','Accounting',8870),
('Carlos Salazar','Accounting',11472),
('Arnav Desai','Accounting',6627),
('Saanvi Sarkar','IT',8113),
('Shirley Rodriguez','IT',5186),
('Nikki Wolf','Sales',9181),
('Alejandro Rosalez','Sales',9441),
('Nikhil Jayashankar','Sales',6660),
('Richard Roe','Sales',10563),
('Pat Candella','SCM',10449),
('Gerard Hernandez','SCM',6949),
('Pamela Castillo','SCM',11303),
('Paulo Santos','SCM',11798),
('Jorge Souza','SCM',10586)
AS basic_pays(employee_name, department, salary);
SELECT * FROM basic_pays;
+-------------------+----------+------+
|    employee_name  |department|salary|
+-------------------+----------+------+
| Arnav Desai       |Accounting|  6627|
```

```
| Jorge Souza       |       SCM| 10586|
| Jane Doe          |Accounting|  8435|
| Nikhil Jayashankar|     Sales|  6660|
| Diego Vanauf      |     Sales| 10563|
| Carlos Salazar    |Accounting| 11472|
| Gerard Hernandez  |       SCM|  6949|
| John Doe          |Accounting|  8992|
| Nikki Wolf        |     Sales|  9181|
| Paulo Santos      |       SCM| 11798|
| Saanvi Sarkar     |        IT|  8113|
| Shirley Rodriguez |        IT|  5186|
| Pat Candella      |       SCM| 10449|
| Akua Mansa        |Accounting|  9998|
| Pamela Castillo   |       SCM| 11303|
| Alejandro Rosalez |     Sales|  9441|
| Juan Li           |Accounting|  8870|
+-------------------+----------+------+
SELECT
department,
percentile_cont(0.25) WITHIN GROUP (ORDER BY salary) AS pc1,
percentile_cont(0.25) WITHIN GROUP (ORDER BY salary) FILTER (WHERE employee_name LIKE
 '%Bo%') AS pc2,
percentile_cont(0.25) WITHIN GROUP (ORDER BY salary DESC) AS pc3,
percentile_cont(0.25) WITHIN GROUP (ORDER BY salary DESC) FILTER (WHERE employee_name
 LIKE '%Bo%') AS pc4,
percentile_disc(0.25) WITHIN GROUP (ORDER BY salary) AS pd1,
percentile_disc(0.25) WITHIN GROUP (ORDER BY salary) FILTER (WHERE employee_name LIKE
 '%Bo%') AS pd2,
percentile_disc(0.25) WITHIN GROUP (ORDER BY salary DESC) AS pd3,
percentile_disc(0.25) WITHIN GROUP (ORDER BY salary DESC) FILTER (WHERE employee_name
 LIKE '%Bo%') AS pd4
FROM basic_pays
GROUP BY department
ORDER BY department;
+----------+-------+--------+-------+--------+-----+-----+-----+-----+
|department|    pc1|     pc2|    pc3|     pc4|  pd1|  pd2|  pd3|  pd4|
+----------+-------+--------+-------+--------+-----+-----+-----+-----+
|Accounting|8543.75| 7838.25| 9746.5|10260.75| 8435| 6627| 9998|11472|
|        IT|5917.75|    NULL|7381.25|    NULL| 5186| NULL| 8113| NULL|
|     Sales|8550.75|    NULL| 9721.5|    NULL| 6660| NULL|10563| NULL|
|       SCM|10449.0|10786.25|11303.0|11460.75|10449|10449|11303|11798|
+----------+-------+--------+-------+--------+-----+-----+-----+-----+
```

Funciones condicionales

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando SQL, consulte the section called “Comandos SQL compatibles”.

Función	Descripción
coalesce(expr1, expr2, ...)	Devuelve el primer argumento no nulo si existe. De lo contrario, es nulo.
if(expr1, expr2, expr3)	Si se evalúa `expr1` como verdadero, devuelve `expr2`; de lo contrario, devuelve `expr3`.
ifnull(expr1, expr2)	Devuelve `expr2` si `expr1` es nulo, o `expr1`.
nanvl(expr1, expr2)	Devuelve `expr1` si no es NaN, o `expr2`.
nullif(expr1, expr2)	Devuelve null si `expr1` es igual a `expr2`, si no, devuelve `expr1`.
nvl(expr1, expr2)	Devuelve `expr2` si `expr1` es nulo, o `expr1`.
nvl2(expr1, expr2, expr3)	Devuelve `expr2` si `expr1` no es nulo, o `expr3`.
CASE WHEN expr1 THEN expr2 [WHEN expr3 THEN expr4]* [ELSE expr5] END	Cuando `expr1` es true, devuelve `expr2`; si no, cuando `expr3` es true, devuelve `expr4`; si no, devuelve `expr5`.

Ejemplos

```
-- coalesce
SELECT coalesce(NULL, 1, NULL);
+----------------------+
|coalesce(NULL, 1, NULL)|
+----------------------+
```

```
|                     1|
+----------------------+
-- if
SELECT if(1 < 2, 'a', 'b');
+------------------+
|(IF((1 < 2), a, b))|
+------------------+
|                 a|
+------------------+
-- ifnull
SELECT ifnull(NULL, array('2'));
+----------------------+
|ifnull(NULL, array(2))|
+----------------------+
|                   [2]|
+----------------------+
-- nanvl
SELECT nanvl(cast('NaN' as double), 123);
+-------------------------------+
|nanvl(CAST(NaN AS DOUBLE), 123)|
+-------------------------------+
|                          123.0|
+-------------------------------+
-- nullif
SELECT nullif(2, 2);
+------------+
|nullif(2, 2)|
+------------+
|        NULL|
+------------+
-- nvl
SELECT nvl(NULL, array('2'));
+------------------+
|nvl(NULL, array(2))|
+------------------+
|               [2]|
+------------------+
-- nvl2
SELECT nvl2(NULL, 2, 1);
+----------------+
|nvl2(NULL, 2, 1)|
+----------------+
|               1|
+----------------+
```

```
-- when
SELECT CASE WHEN 1 > 0 THEN 1 WHEN 2 > 0 THEN 2.0 ELSE 1.2 END;
+-------------------------------------------------------------+
|CASE WHEN (1 > 0) THEN 1 WHEN (2 > 0) THEN 2.0 ELSE 1.2 END|
+-------------------------------------------------------------+
|                                                          1.0|
+-------------------------------------------------------------+
SELECT CASE WHEN 1 < 0 THEN 1 WHEN 2 > 0 THEN 2.0 ELSE 1.2 END;
+-------------------------------------------------------------+
|CASE WHEN (1 < 0) THEN 1 WHEN (2 > 0) THEN 2.0 ELSE 1.2 END|
+-------------------------------------------------------------+
|                                                          2.0|
+-------------------------------------------------------------+
SELECT CASE WHEN 1 < 0 THEN 1 WHEN 2 < 0 THEN 2.0 END;
+-------------------------------------------------+
|CASE WHEN (1 < 0) THEN 1 WHEN (2 < 0) THEN 2.0 END|
+-------------------------------------------------+
|                                             NULL|
+-------------------------------------------------+
```

Funciones JSON

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando SQL, consulte the section called “Comandos SQL compatibles”.

Función	Descripción
from_json (jsonStr, schema[, options])	Devuelve un valor de estructura con los valores `jsonStr` y `schema` dados.
get_json_ object(js on_txt, path)	Extrae un objeto json de `path`.

Función	Descripción
json_array_length(jsonArray)	Devuelve la cantidad de elementos en la matriz exterior de una cadena JSON.
json_object_keys(json_object)	Devuelve todas las claves del objeto JSON más externo como una matriz.
json_tuple(jsonStr, p1, p2, ..., pn)	Devuelve una tupla como la función get_json_object, pero toma varios nombres. Todos los parámetros de entrada y los tipos de columnas de salida son cadenas.
schema_of_json(json[, options])	Devuelve el esquema en el formato DDL de la cadena JSON.
to_json(expr[, options])	Devuelve una cadena JSON con un valor de estructura determinado

Ejemplos

```
-- from_json
SELECT from_json('{"a":1, "b":0.8}', 'a INT, b DOUBLE');
+--------------------------+
| from_json({"a":1, "b":0.8}) |
+--------------------------+
| {1, 0.8}                 |
+--------------------------+

SELECT from_json('{"time":"26/08/2015"}', 'time Timestamp', map('timestampFormat', 'dd/
MM/yyyy'));
+--------------------------------+
| from_json({"time":"26/08/2015"}) |
+--------------------------------+
| {2015-08-26 00:00...           |
+--------------------------------+
```

```
SELECT from_json('{"teacher": "Alice", "student": [{"name": "Bob", "rank": 1}, {"name":
 "Charlie", "rank": 2}]}', 'STRUCT<teacher: STRING, student: ARRAY<STRUCT<name: STRING,
 rank: INT>>>');
+-----------------------------------------------------------------------------------------------
+
| from_json({"teacher": "Alice", "student": [{"name": "Bob", "rank": 1}, {"name":
 "Charlie", "rank": 2}]}) |
+-----------------------------------------------------------------------------------------------
+
| {Alice, [{Bob, 1}...
                  |
+-----------------------------------------------------------------------------------------------
+

-- get_json_object
SELECT get_json_object('{"a":"b"}', '$.a');
+-------------------------------+
| get_json_object({"a":"b"}, $.a) |
+-------------------------------+
| b                             |
+-------------------------------+

-- json_array_length
SELECT json_array_length('[1,2,3,4]');
+----------------------------+
| json_array_length([1,2,3,4]) |
+----------------------------+
| 4                          |
+----------------------------+

SELECT json_array_length('[1,2,3,{"f1":1,"f2":[5,6]},4]');
+-------------------------------------------------+
| json_array_length([1,2,3,{"f1":1,"f2":[5,6]},4]) |
+-------------------------------------------------+
| 5                                               |
+-------------------------------------------------+

SELECT json_array_length('[1,2');
+----------------------+
| json_array_length([1,2) |
+----------------------+
| NULL                 |
+----------------------+
```

```
-- json_object_keys
SELECT json_object_keys('{}');
+--------------------+
| json_object_keys({}) |
+--------------------+
| []                 |
+--------------------+

SELECT json_object_keys('{"key": "value"}');
+---------------------------------+
| json_object_keys({"key": "value"}) |
+---------------------------------+
| [key]                           |
+---------------------------------+

SELECT json_object_keys('{"f1":"abc","f2":{"f3":"a", "f4":"b"}}');
+-----------------------------------------------------------+
| json_object_keys({"f1":"abc","f2":{"f3":"a", "f4":"b"}}) |
+-----------------------------------------------------------+
| [f1, f2]                                                  |
+-----------------------------------------------------------+

-- json_tuple
SELECT json_tuple('{"a":1, "b":2}', 'a', 'b');
+---+---+
| c0| c1|
+---+---+
|  1|  2|
+---+---+

-- schema_of_json
SELECT schema_of_json('[{"col":0}]');
+---------------------------+
| schema_of_json([{"col":0}]) |
+---------------------------+
| ARRAY<STRUCT<col:...      |
+---------------------------+

SELECT schema_of_json('[{"col":01}]', map('allowNumericLeadingZeros', 'true'));
+---------------------------+
| schema_of_json([{"col":01}]) |
+---------------------------+
| ARRAY<STRUCT<col:...       |
+---------------------------+
```

```
-- to_json
SELECT to_json(named_struct('a', 1, 'b', 2));
+--------------------------------+
| to_json(named_struct(a, 1, b, 2)) |
+--------------------------------+
| {"a":1,"b":2}                  |
+--------------------------------+

SELECT to_json(named_struct('time', to_timestamp('2015-08-26', 'yyyy-MM-dd')),
 map('timestampFormat', 'dd/MM/yyyy'));
+-------------------------------------------------------------------+
| to_json(named_struct(time, to_timestamp(2015-08-26, yyyy-MM-dd))) |
+-------------------------------------------------------------------+
| {"time":"26/08/20...                                              |
+-------------------------------------------------------------------+

SELECT to_json(array(named_struct('a', 1, 'b', 2)));
+----------------------------------------+
| to_json(array(named_struct(a, 1, b, 2))) |
+----------------------------------------+
| [{"a":1,"b":2}]                        |
+----------------------------------------+

SELECT to_json(map('a', named_struct('b', 1)));
+-----------------------------------+
| to_json(map(a, named_struct(b, 1))) |
+-----------------------------------+
| {"a":{"b":1}}                     |
+-----------------------------------+

SELECT to_json(map(named_struct('a', 1),named_struct('b', 2)));
+-----------------------------------------------------+
| to_json(map(named_struct(a, 1), named_struct(b, 2))) |
+-----------------------------------------------------+
| {"[1]":{"b":2}}                                     |
+-----------------------------------------------------+

SELECT to_json(map('a', 1));
+------------------+
| to_json(map(a, 1)) |
+------------------+
| {"a":1}          |
+------------------+
```

```
SELECT to_json(array(map('a', 1)));
+-------------------------+
| to_json(array(map(a, 1))) |
+-------------------------+
| [{"a":1}]               |
+-------------------------+
```

Funciones de matriz

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando SQL, consulte the section called “Comandos SQL compatibles”.

Función	Descripción
array(expr, ...)	Devuelve una matriz con los elementos dados.
array_append(array, element)	Agrega el elemento al final de la matriz pasada como primer argumento. El tipo de elemento debe ser similar al tipo de los elementos de la matriz. El elemento nulo también se agrega a la matriz. Pero si la matriz se pasa, es NULL, la salida es NULL
array_compact(array)	Elimina los valores nulos de la matriz.
array_contains(array, value)	Devuelve verdadero si la matriz contiene el valor.
array_distinct(array)	Elimina los valores duplicados de la matriz.
array_except(array1, array2)	Devuelve una matriz de los elementos de la array1 pero no de la array2, sin duplicados.
array_insert(x, pos, val)	Coloca val en el punto de índice de la matriz x. Los índices comienzan en 1. El índice negativo máximo es -1 para el que la función inserta un

Función	Descripción
	nuevo elemento después del último elemento actual. El índice por encima del tamaño de la matriz se agrega a la matriz, o antepone la matriz si el índice es negativo, con elementos “nulos”.
array_intersect(array1, array2)	Devuelve una matriz de los elementos en la intersección de array1 y array2, sin duplicados.
array_join(array, delimiter[, nullReplacement])	Concatena los elementos de la matriz dada mediante el delimitador y una cadena opcional para reemplazar los valores nulos. Si no se establece ningún valor para nullReplacement, se filtra cualquier valor nulo.
array_max(array)	Devuelve el valor máximo de la matriz. NaN es mayor que cualquier elemento que no sea NaN para el tipo doble/flotante. Se omiten los elementos NULL.
array_min(array)	Devuelve el valor mínimo de la matriz. NaN es mayor que cualquier elemento que no sea NaN para el tipo doble/flotante. Se omiten los elementos NULL.
array_position(array, element)	Devuelve el índice (basado en 1) del primer elemento coincidente de la matriz siempre que sea largo, o 0 si no se encuentra ninguna coincidencia.

Función	Descripción
array_prepend(array, element)	Agrega el elemento al principio de la matriz pasada como primer argumento. El tipo de elemento debe ser el mismo que el tipo de los elementos de la matriz. El elemento nulo también se antepone a la matriz. Pero si la matriz pasada era NULL, la salida va a ser NULL
array_remove(array, element)	Elimine todos los elementos que sean iguales al elemento de la matriz.
array_repeat(element, count)	Devuelve la matriz que contiene los tiempos de recuento de elementos.
array_union(array1, array2)	Devuelve una matriz de los elementos de la unión de array1 y array2, sin duplicados.
arrays_overlap(a1, a2)	Devuelve verdadero si a1 contiene al menos un elemento no nulo presente también en a2. Si las matrices no tienen ningún elemento común, ninguna está vacía y alguna de ellas contiene un elemento nulo, se devuelve null y falso en caso contrario.
arrays_zip(a1, a2, ...)	Devuelve una matriz combinada de estructur as en la que la estructura n-ésima contiene todos los valores n-ésimos de las matrices de entrada.
flatten(arrayOfArrays)	Transforma una matriz de matrices en una sola matriz.
get(array, index)	Devuelve el elemento de la matriz en un índice dado (basado en 0). Si el índice apunta fuera de los límites de la matriz, esta función devuelve NULL.

Función	Descripción
sequence(start, stop, step)	Genera una matriz de elementos desde el principio hasta la parada (ambos incluidos), incrementándose paso a paso. El tipo de valor devuelto es el mismo que el tipo de la expresión. Los tipos compatibles son byte, short, integer, long, date, timestamp. Las expresiones de inicio y parada deben tener el mismo tipo de resolución. Si las expresiones de inicio y finalización se resuelven en el tipo “date” o “timestamp”, la expresión escalonada debe resolverse en el tipo “interval”, “year-mon th interval” o “day-time interval”; de lo contrario , debe ser del mismo tipo que las expresiones de inicio y finalización.
shuffle(array)	Devuelve una permutación aleatoria de la matriz dada.
slice(x, start, length)	Subestablece la matriz x a partir del inicio del índice (los índices de la matriz comienzan en 1 o comienzan desde el final si el inicio es negativo) con la longitud especificada.
sort_array(array[, ascendingOrder])	Ordena la matriz de entrada en orden ascendente o descendente de acuerdo con el orden natural de los elementos de la matriz. NaN es mayor que cualquier elemento que no sea NaN para el tipo doble/flotante. Los elementos nulos se colocarán al principio de la matriz devuelta en orden ascendente o al final de la matriz devuelta en orden descendente.

Ejemplos

```
-- array
```

```
SELECT array(1, 2, 3);
+--------------+
|array(1, 2, 3)|
+--------------+
|     [1, 2, 3]|
+--------------+
-- array_append
SELECT array_append(array('b', 'd', 'c', 'a'), 'd');
+----------------------------------+
|array_append(array(b, d, c, a), d)|
+----------------------------------+
|                   [b, d, c, a, d]|
+----------------------------------+
SELECT array_append(array(1, 2, 3, null), null);
+---------------------------------------+
|array_append(array(1, 2, 3, NULL), NULL)|
+---------------------------------------+
|                    [1, 2, 3, NULL, N...|
+---------------------------------------+
SELECT array_append(CAST(null as Array<Int>), 2);
+---------------------+
|array_append(NULL, 2)|
+---------------------+
|                 NULL|
+---------------------+
-- array_compact
SELECT array_compact(array(1, 2, 3, null));
+----------------------------------+
|array_compact(array(1, 2, 3, NULL))|
+----------------------------------+
|                          [1, 2, 3]|
+----------------------------------+
SELECT array_compact(array("a", "b", "c"));
+----------------------------+
|array_compact(array(a, b, c))|
+----------------------------+
|                    [a, b, c]|
+----------------------------+
-- array_contains
SELECT array_contains(array(1, 2, 3), 2);
+---------------------------------+
|array_contains(array(1, 2, 3), 2)|
+---------------------------------+
|                             true|
```

```
+--------------------------------+
-- array_distinct
SELECT array_distinct(array(1, 2, 3, null, 3));
+--------------------------------------+
|array_distinct(array(1, 2, 3, NULL, 3))|
+--------------------------------------+
|                        [1, 2, 3, NULL]|
+--------------------------------------+
-- array_except
SELECT array_except(array(1, 2, 3), array(1, 3, 5));
+-------------------------------------------+
|array_except(array(1, 2, 3), array(1, 3, 5))|
+-------------------------------------------+
|                                        [2]|
+-------------------------------------------+
-- array_insert
SELECT array_insert(array(1, 2, 3, 4), 5, 5);
+-----------------------------------+
|array_insert(array(1, 2, 3, 4), 5, 5)|
+-----------------------------------+
|                     [1, 2, 3, 4, 5]|
+-----------------------------------+
SELECT array_insert(array(5, 4, 3, 2), -1, 1);
+-----------------------------------+
|array_insert(array(5, 4, 3, 2), -1, 1)|
+-----------------------------------+
|                      [5, 4, 3, 2, 1]|
+-----------------------------------+
SELECT array_insert(array(5, 3, 2, 1), -4, 4);
+-----------------------------------+
|array_insert(array(5, 3, 2, 1), -4, 4)|
+-----------------------------------+
|                      [5, 4, 3, 2, 1]|
+-----------------------------------+
-- array_intersect
SELECT array_intersect(array(1, 2, 3), array(1, 3, 5));
+-----------------------------------------------+
|array_intersect(array(1, 2, 3), array(1, 3, 5))|
+-----------------------------------------------+
|                                         [1, 3]|
+-----------------------------------------------+
-- array_join
SELECT array_join(array('hello', 'world'), ' ');
+--------------------------------+
```

```
|array_join(array(hello, world),  )|
+----------------------------------+
|                       hello world|
+----------------------------------+
SELECT array_join(array('hello', null ,'world'), ' ');
+---------------------------------------+
|array_join(array(hello, NULL, world),  )|
+---------------------------------------+
|                             hello world|
+---------------------------------------+
SELECT array_join(array('hello', null ,'world'), ' ', ',');
+-------------------------------------------+
|array_join(array(hello, NULL, world),  , ,)|
+-------------------------------------------+
|                              hello , world|
+-------------------------------------------+
-- array_max
SELECT array_max(array(1, 20, null, 3));
+--------------------------------+
|array_max(array(1, 20, NULL, 3))|
+--------------------------------+
|                              20|
+--------------------------------+
-- array_min
SELECT array_min(array(1, 20, null, 3));
+--------------------------------+
|array_min(array(1, 20, NULL, 3))|
+--------------------------------+
|                               1|
+--------------------------------+
-- array_position
SELECT array_position(array(312, 773, 708, 708), 708);
+-----------------------------------------------+
|array_position(array(312, 773, 708, 708), 708)|
+-----------------------------------------------+
|                                              3|
+-----------------------------------------------+
SELECT array_position(array(312, 773, 708, 708), 414);
+-----------------------------------------------+
|array_position(array(312, 773, 708, 708), 414)|
+-----------------------------------------------+
|                                              0|
+-----------------------------------------------+
-- array_prepend
```

```
SELECT array_prepend(array('b', 'd', 'c', 'a'), 'd');
+----------------------------------+
|array_prepend(array(b, d, c, a), d)|
+----------------------------------+
|                    [d, b, d, c, a]|
+----------------------------------+
SELECT array_prepend(array(1, 2, 3, null), null);
+---------------------------------------+
|array_prepend(array(1, 2, 3, NULL), NULL)|
+---------------------------------------+
|                     [NULL, 1, 2, 3, N...|
+---------------------------------------+
SELECT array_prepend(CAST(null as Array<Int>), 2);
+----------------------+
|array_prepend(NULL, 2)|
+----------------------+
|                  NULL|
+----------------------+
-- array_remove
SELECT array_remove(array(1, 2, 3, null, 3), 3);
+---------------------------------------+
|array_remove(array(1, 2, 3, NULL, 3), 3)|
+---------------------------------------+
|                           [1, 2, NULL]|
+---------------------------------------+
-- array_repeat
SELECT array_repeat('123', 2);
+--------------------+
|array_repeat(123, 2)|
+--------------------+
|          [123, 123]|
+--------------------+
-- array_union
SELECT array_union(array(1, 2, 3), array(1, 3, 5));
+-----------------------------------------+
|array_union(array(1, 2, 3), array(1, 3, 5))|
+-----------------------------------------+
|                               [1, 2, 3, 5]|
+-----------------------------------------+
-- arrays_overlap
SELECT arrays_overlap(array(1, 2, 3), array(3, 4, 5));
+-------------------------------------------+
|arrays_overlap(array(1, 2, 3), array(3, 4, 5))|
+-------------------------------------------+
```

```
|                                             true|
+-------------------------------------------------+
-- arrays_zip
SELECT arrays_zip(array(1, 2, 3), array(2, 3, 4));
+-----------------------------------------+
|arrays_zip(array(1, 2, 3), array(2, 3, 4))|
+-----------------------------------------+
|                      [{1, 2}, {2, 3}, ...|
+-----------------------------------------+
SELECT arrays_zip(array(1, 2), array(2, 3), array(3, 4));
+-------------------------------------------------+
|arrays_zip(array(1, 2), array(2, 3), array(3, 4))|
+-------------------------------------------------+
|                             [{1, 2, 3}, {2, 3...|
+-------------------------------------------------+
-- flatten
SELECT flatten(array(array(1, 2), array(3, 4)));
+---------------------------------------+
|flatten(array(array(1, 2), array(3, 4)))|
+---------------------------------------+
|                           [1, 2, 3, 4]|
+---------------------------------------+
-- get
SELECT get(array(1, 2, 3), 0);
+----------------------+
|get(array(1, 2, 3), 0)|
+----------------------+
|                     1|
+----------------------+
SELECT get(array(1, 2, 3), 3);
+----------------------+
|get(array(1, 2, 3), 3)|
+----------------------+
|                  NULL|
+----------------------+
SELECT get(array(1, 2, 3), -1);
+-----------------------+
|get(array(1, 2, 3), -1)|
+-----------------------+
|                   NULL|
+-----------------------+
-- sequence
SELECT sequence(1, 5);
+---------------+
```

```
| sequence(1, 5)|
+---------------+
|[1, 2, 3, 4, 5]|
+---------------+
SELECT sequence(5, 1);
+---------------+
| sequence(5, 1)|
+---------------+
|[5, 4, 3, 2, 1]|
+---------------+
SELECT sequence(to_date('2018-01-01'), to_date('2018-03-01'), interval 1 month);
+------------------------------------------------------------------------+
|sequence(to_date(2018-01-01), to_date(2018-03-01), INTERVAL '1' MONTH)|
+------------------------------------------------------------------------+
|                                                    [2018-01-01, 2018...|
+------------------------------------------------------------------------+
SELECT sequence(to_date('2018-01-01'), to_date('2018-03-01'), interval '0-1' year to
 month);
+-----------------------------------------------------------------------------------+
|sequence(to_date(2018-01-01), to_date(2018-03-01), INTERVAL '0-1' YEAR TO MONTH)|
+-----------------------------------------------------------------------------------+
|                                                                 [2018-01-01, 2018...|
+-----------------------------------------------------------------------------------+
-- shuffle
SELECT shuffle(array(1, 20, 3, 5));
+---------------------------+
|shuffle(array(1, 20, 3, 5))|
+---------------------------+
|              [5, 1, 20, 3]|
+---------------------------+
SELECT shuffle(array(1, 20, null, 3));
+-----------------------------+
|shuffle(array(1, 20, NULL, 3))|
+-----------------------------+
|              [1, NULL, 20, 3]|
+-----------------------------+
-- slice
SELECT slice(array(1, 2, 3, 4), 2, 2);
+-----------------------------+
|slice(array(1, 2, 3, 4), 2, 2)|
+-----------------------------+
|                        [2, 3]|
+-----------------------------+
SELECT slice(array(1, 2, 3, 4), -2, 2);
```

```
+------------------------------+
|slice(array(1, 2, 3, 4), -2, 2)|
+------------------------------+
|                        [3, 4]|
+------------------------------+
-- sort_array
SELECT sort_array(array('b', 'd', null, 'c', 'a'), true);
+---------------------------------------+
|sort_array(array(b, d, NULL, c, a), true)|
+---------------------------------------+
|                      [NULL, a, b, c, d]|
+---------------------------------------+
```

Funciones de ventana

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando SQL, consulte the section called “Comandos SQL compatibles”.

Las funciones de ventana funcionan en un grupo de filas, denominado ventana y calculan un valor de devolución para cada fila en función del grupo de filas. Las funciones de ventana son útiles para procesar tareas como calcular un promedio móvil, computar una estadística acumulada o acceder al valor de las filas dada la posición relativa de la fila actual.

Sintaxis

```
window_function [ nulls_option ] OVER ( [ { PARTITION | DISTRIBUTE } BY
 partition_col_name = partition_col_val ( [ , ... ] ) ] { ORDER | SORT } BY expression
 [ ASC | DESC ] [ NULLS { FIRST | LAST } ] [ , ... ] [ window_frame ] )
```

Parámetros

- Funciones de clasificación

  Sintaxis: `RANK | DENSE_RANK | PERCENT_RANK | NTILE | ROW_NUMBER`

  Funciones analíticas

  Sintaxis: `CUME_DIST | LAG | LEAD | NTH_VALUE | FIRST_VALUE | LAST_VALUE`

Funciones de agregación

Sintaxis: `MAX | MIN | COUNT | SUM | AVG | ...`

- `nulls_option`: Especifica si se deben omitir o no los valores nulos al evaluar la función de ventana. RESPECT NULLS significa no omitir los valores nulos, mientras que IGNORE NULLS significa omitirlos. Si no se especifica, el valor predeterminado es RESPECT NULLS.

  Sintaxis: `{ IGNORE | RESPECT } NULLS`

  Nota: `Only LAG | LEAD | NTH_VALUE | FIRST_VALUE | LAST_VALUE` se pueden utilizar con `IGNORE NULLS`.
- `window_frame`: Especifica en qué fila se debe iniciar la ventana y dónde se debe terminar.

  Sintaxis: `{ RANGE | ROWS } { frame_start | BETWEEN frame_start AND frame_end }`

  frame_start y frame_end tienen la siguiente sintaxis:

  Sintaxis: `UNBOUNDED PRECEDING | offset PRECEDING | CURRENT ROW | offset FOLLOWING | UNBOUNDED FOLLOWING`

  offset: especifica el desplazamiento desde la posición de la fila actual.

  Nota: Si se omite frame_end, el valor predeterminado es CURRENT ROW.

Ejemplos

```
CREATE TABLE employees (name STRING, dept STRING, salary INT, age INT);
INSERT INTO employees VALUES ("Lisa", "Sales", 10000, 35);
INSERT INTO employees VALUES ("Evan", "Sales", 32000, 38);
INSERT INTO employees VALUES ("Fred", "Engineering", 21000, 28);
INSERT INTO employees VALUES ("Alex", "Sales", 30000, 33);
INSERT INTO employees VALUES ("Tom", "Engineering", 23000, 33);
INSERT INTO employees VALUES ("Jane", "Marketing", 29000, 28);
INSERT INTO employees VALUES ("Jeff", "Marketing", 35000, 38);
INSERT INTO employees VALUES ("Paul", "Engineering", 29000, 23);
INSERT INTO employees VALUES ("Chloe", "Engineering", 23000, 25);
SELECT * FROM employees;
+-----+-----------+------+-----+
| name|       dept|salary|  age|
```

```
+-----+-----------+------+-----+
|Chloe|Engineering| 23000|   25|
| Fred|Engineering| 21000|   28|
| Paul|Engineering| 29000|   23|
|Helen|  Marketing| 29000|   40|
|  Tom|Engineering| 23000|   33|
| Jane|  Marketing| 29000|   28|
| Jeff|  Marketing| 35000|   38|
| Evan|      Sales| 32000|   38|
| Lisa|      Sales| 10000|   35|
| Alex|      Sales| 30000|   33|
+-----+-----------+------+-----+
SELECT name, dept, salary, RANK() OVER (PARTITION BY dept ORDER BY salary) AS rank FROM
 employees;
+-----+-----------+------+----+
| name|       dept|salary|rank|
+-----+-----------+------+----+
| Lisa|      Sales| 10000|   1|
| Alex|      Sales| 30000|   2|
| Evan|      Sales| 32000|   3|
| Fred|Engineering| 21000|   1|
|  Tom|Engineering| 23000|   2|
|Chloe|Engineering| 23000|   2|
| Paul|Engineering| 29000|   4|
|Helen|  Marketing| 29000|   1|
| Jane|  Marketing| 29000|   1|
| Jeff|  Marketing| 35000|   3|
+-----+-----------+------+----+
SELECT name, dept, salary, DENSE_RANK() OVER (PARTITION BY dept ORDER BY salary ROWS
 BETWEEN
UNBOUNDED PRECEDING AND CURRENT ROW) AS dense_rank FROM employees;
+-----+-----------+------+----------+
| name|       dept|salary|dense_rank|
+-----+-----------+------+----------+
| Lisa|      Sales| 10000|         1|
| Alex|      Sales| 30000|         2|
| Evan|      Sales| 32000|         3|
| Fred|Engineering| 21000|         1|
|  Tom|Engineering| 23000|         2|
|Chloe|Engineering| 23000|         2|
| Paul|Engineering| 29000|         3|
|Helen|  Marketing| 29000|         1|
| Jane|  Marketing| 29000|         1|
| Jeff|  Marketing| 35000|         2|
```

```
+-----+-----------+------+----------+
SELECT name, dept, age, CUME_DIST() OVER (PARTITION BY dept ORDER BY age
RANGE BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW) AS cume_dist FROM employees;
+-----+-----------+------+------------------+
| name|       dept|age   |         cume_dist|
+-----+-----------+------+------------------+
| Alex|      Sales|    33|0.3333333333333333|
| Lisa|      Sales|    35|0.6666666666666666|
| Evan|      Sales|    38|               1.0|
| Paul|Engineering|    23|              0.25|
|Chloe|Engineering|    25|              0.75|
| Fred|Engineering|    28|              0.25|
|  Tom|Engineering|    33|               1.0|
| Jane|  Marketing|    28|0.3333333333333333|
| Jeff|  Marketing|    38|0.6666666666666666|
|Helen|  Marketing|    40|               1.0|
+-----+-----------+------+------------------+
SELECT name, dept, salary, MIN(salary) OVER (PARTITION BY dept ORDER BY salary) AS min
FROM employees;
+-----+-----------+------+-----+
| name|       dept|salary|  min|
+-----+-----------+------+-----+
| Lisa|      Sales| 10000|10000|
| Alex|      Sales| 30000|10000|
| Evan|      Sales| 32000|10000|
|Helen|  Marketing| 29000|29000|
| Jane|  Marketing| 29000|29000|
| Jeff|  Marketing| 35000|29000|
| Fred|Engineering| 21000|21000|
|  Tom|Engineering| 23000|21000|
|Chloe|Engineering| 23000|21000|
| Paul|Engineering| 29000|21000|
+-----+-----------+------+-----+
SELECT name, salary,
LAG(salary) OVER (PARTITION BY dept ORDER BY salary) AS lag,
LEAD(salary, 1, 0) OVER (PARTITION BY dept ORDER BY salary) AS lead
FROM employees;
+-----+-----------+------+-----+-----+
| name|       dept|salary|  lag| lead|
+-----+-----------+------+-----+-----+
| Lisa|      Sales| 10000|NULL |30000|
| Alex|      Sales| 30000|10000|32000|
| Evan|      Sales| 32000|30000|    0|
| Fred|Engineering| 21000| NULL|23000|
```

```
|Chloe|Engineering| 23000|21000|23000|
|  Tom|Engineering| 23000|23000|29000|
| Paul|Engineering| 29000|23000|    0|
|Helen|  Marketing| 29000| NULL|29000|
| Jane|  Marketing| 29000|29000|35000|
| Jeff|  Marketing| 35000|29000|    0|
+-----+-----------+------+-----+-----+
SELECT id, v,
LEAD(v, 0) IGNORE NULLS OVER w lead,
LAG(v, 0) IGNORE NULLS OVER w lag,
NTH_VALUE(v, 2) IGNORE NULLS OVER w nth_value,
FIRST_VALUE(v) IGNORE NULLS OVER w first_value,
LAST_VALUE(v) IGNORE NULLS OVER w last_value
FROM test_ignore_null
WINDOW w AS (ORDER BY id)
ORDER BY id;
+--+----+----+----+---------+-----------+----------+
|id|   v|lead| lag|nth_value|first_value|last_value|
+--+----+----+----+---------+-----------+----------+
| 0|NULL|NULL|NULL|     NULL|       NULL|      NULL|
| 1|   x|   x|   x|     NULL|          x|         x|
| 2|NULL|NULL|NULL|     NULL|          x|         x|
| 3|NULL|NULL|NULL|     NULL|          x|         x|
| 4|   y|   y|   y|        y|          x|         y|
| 5|NULL|NULL|NULL|        y|          x|         y|
| 6|   z|   z|   z|        y|          x|         z|
| 7|   v|   v|   v|        y|          x|         v|
| 8|NULL|NULL|NULL|        y|          x|         v|
+--+----+----+----+---------+-----------+----------+
```

Funciones de conversión

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando SQL, consulte the section called “Comandos SQL compatibles”.

Función	Descripción
bigint(expr)	Transfiere el valor `expr` al tipo de datos de destino `bigint`.

Función	Descripción
binary(expr)	Transfiere el valor `expr` al tipo de datos de destino `binario`.
boolean(expr)	Transfiere el valor `expr` al tipo de datos de destino `boolean`.
cast(expr AS type)	Convierte el valor `expr` al tipo de datos de destino `type`.
date(expr)	Transfiere el valor `expr` al tipo de datos de destino `date`.
decimal(expr)	Convierte el valor `expr` al tipo de datos de destino `decimal`.
double(expr)	Convierte el valor `expr` al tipo de datos de destino `double`.
float(expr)	Convierte el valor `expr` al tipo de datos de destino `float`.
int(expr)	Transfiere el valor `expr` al tipo de datos de destino `int`.
smallint(expr)	Transfiere el valor `expr` al tipo de datos de destino `smallint`.
string(expr)	Transfiere el valor `expr` al tipo de datos de destino `string`.
timestamp(expr)	Transfiere el valor `expr` al tipo de datos de destino `timestamp`.
tinyint(expr)	Transfiere el valor `expr` al tipo de datos de destino `tinyint`.

Ejemplos

```
-- cast
SELECT cast(field as int);
+---------------+
|CAST(field AS INT)|
+---------------+
|             10|
+---------------+
```

Funciones de predicados

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando SQL, consulte the section called “Comandos SQL compatibles”.

Función	Descripción
! expr	No lógico.
expr1 < expr2	Devuelve el valor verdadero si `expr1` es menor que `expr2`.
expr1 <= expr2	Devuelve verdadero si `expr1` es menor o igual que `expr2`.
expr1 <=> expr2	Devuelve el mismo resultado que el operador EQUAL (=) para los operandos no nulos, pero devuelve verdadero si ambos son nulos y falso si uno de ellos es nulo.
expr1 = expr2	Devuelve verdadero si `expr1` es igual a `expr2`, o falso en caso contrario.
expr1 == expr2	Devuelve verdadero si `expr1` es igual a `expr2`, o falso en caso contrario.
expr1 > expr2	Devuelve verdadero si `expr1` es mayor que `expr2`.
expr1 >= expr2	Devuelve verdadero si `expr1` es mayor o igual que `expr2`.
expr1 y expr2	AND lógico.
str ilike pattern[ESCAPE escape]	Devuelve verdadero si str hace coincidir `pattern` con `escape` sin distinguir mayúsculas de minúsculas, null si algún argumento es nulo, falso en caso contrario.
expr1 in(expr2, expr3, ...)	Devuelve verdadero si `expr` es igual a cualquier valN.
isnan(expr)	Devuelve verdadero si `expr` es NaN, o falso en caso contrario.

Función	Descripción
isnotnull(expr)	Devuelve verdadero si `expr` no es nulo, o falso en caso contrario.
isnull(expr)	Devuelve verdadero si `expr` es nulo o falso en caso contrario.
str like pattern[ESCAPE escape]	Devuelve verdadero si str coincide con `pattern` con `escape`, null si algún argumento es nulo, falso en caso contrario.
not expr	No lógico.
expr1 o expr2	OR lógico.
regexp(str, regexp)	Devuelve verdadero si `str` coincide con `regexp`, o falso en caso contrario.
regexp_like(str, regexp)	Devuelve verdadero si `str` coincide con `regexp`, o falso en caso contrario.
rlike(str, regexp)	Devuelve verdadero si `str` coincide con `regexp`, o falso en caso contrario.

Ejemplos

```
-- !
SELECT ! true;
+----------+
|(NOT true)|
+----------+
|     false|
+----------+
SELECT ! false;
+-----------+
|(NOT false)|
+-----------+
|       true|
+-----------+
SELECT ! NULL;
+----------+
|(NOT NULL)|
+----------+
```

```
|      NULL|
+----------+
-- <
SELECT to_date('2009-07-30 04:17:52') < to_date('2009-07-30 04:17:52');
+------------------------------------------------------------+
|(to_date(2009-07-30 04:17:52) < to_date(2009-07-30 04:17:52))|
+------------------------------------------------------------+
|                                                       false|
+------------------------------------------------------------+
SELECT to_date('2009-07-30 04:17:52') < to_date('2009-08-01 04:17:52');
+------------------------------------------------------------+
|(to_date(2009-07-30 04:17:52) < to_date(2009-08-01 04:17:52))|
+------------------------------------------------------------+
|                                                        true|
+------------------------------------------------------------+
SELECT 1 < NULL;
+----------+
|(1 < NULL)|
+----------+
|      NULL|
+----------+
-- <=
SELECT 2 <= 2;
+--------+
|(2 <= 2)|
+--------+
|    true|
+--------+
SELECT 1.0 <= '1';
+----------+
|(1.0 <= 1)|
+----------+
|      true|
+----------+
SELECT to_date('2009-07-30 04:17:52') <= to_date('2009-07-30 04:17:52');
+------------------------------------------------------------+
|(to_date(2009-07-30 04:17:52) <= to_date(2009-07-30 04:17:52))|
+------------------------------------------------------------+
|                                                        true|
+------------------------------------------------------------+
SELECT to_date('2009-07-30 04:17:52') <= to_date('2009-08-01 04:17:52');
+------------------------------------------------------------+
|(to_date(2009-07-30 04:17:52) <= to_date(2009-08-01 04:17:52))|
+------------------------------------------------------------+
```

```
|                                                                 true|
+-----------------------------------------------------------------+
SELECT 1 <= NULL;
+-----------+
|(1 <= NULL)|
+-----------+
|       NULL|
+-----------+
-- <=>
SELECT 2 <=> 2;
+---------+
|(2 <=> 2)|
+---------+
|     true|
+---------+
SELECT 1 <=> '1';
+---------+
|(1 <=> 1)|
+---------+
|     true|
+---------+
SELECT true <=> NULL;
+---------------+
|(true <=> NULL)|
+---------------+
|          false|
+---------------+
SELECT NULL <=> NULL;
+---------------+
|(NULL <=> NULL)|
+---------------+
|           true|
+---------------+
-- =
SELECT 2 = 2;
+-------+
|(2 = 2)|
+-------+
|   true|
+-------+
SELECT 1 = '1';
+-------+
|(1 = 1)|
+-------+
```

```
|   true|
+-------+
SELECT true = NULL;
+-------------+
|(true = NULL)|
+-------------+
|         NULL|
+-------------+
SELECT NULL = NULL;
+-------------+
|(NULL = NULL)|
+-------------+
|         NULL|
+-------------+
-- ==
SELECT 2 == 2;
+-------+
|(2 = 2)|
+-------+
|   true|
+-------+
SELECT 1 == '1';
+-------+
|(1 = 1)|
+-------+
|   true|
+-------+
SELECT true == NULL;
+-------------+
|(true = NULL)|
+-------------+
|         NULL|
+-------------+
SELECT NULL == NULL;
+-------------+
|(NULL = NULL)|
+-------------+
|         NULL|
+-------------+
-- >
SELECT 2 > 1;
+-------+
|(2 > 1)|
+-------+
```

```
|   true|
+-------+
SELECT 2 > 1.1;
+-------+
|(2 > 1)|
+-------+
|   true|
+-------+
SELECT to_date('2009-07-30 04:17:52') > to_date('2009-07-30 04:17:52');
+-------------------------------------------------------------+
|(to_date(2009-07-30 04:17:52) > to_date(2009-07-30 04:17:52))|
+-------------------------------------------------------------+
|                                                        false|
+-------------------------------------------------------------+
SELECT to_date('2009-07-30 04:17:52') > to_date('2009-08-01 04:17:52');
+-------------------------------------------------------------+
|(to_date(2009-07-30 04:17:52) > to_date(2009-08-01 04:17:52))|
+-------------------------------------------------------------+
|                                                        false|
+-------------------------------------------------------------+
SELECT 1 > NULL;
+----------+
|(1 > NULL)|
+----------+
|      NULL|
+----------+
-- >=
SELECT 2 >= 1;
+--------+
|(2 >= 1)|
+--------+
|    true|
+--------+
SELECT 2.0 >= '2.1';
+------------+
|(2.0 >= 2.1)|
+------------+
|       false|
+------------+
SELECT to_date('2009-07-30 04:17:52') >= to_date('2009-07-30 04:17:52');
+-------------------------------------------------------------+
|(to_date(2009-07-30 04:17:52) >= to_date(2009-07-30 04:17:52))|
+-------------------------------------------------------------+
|                                                        true|
```

```
+-------------------------------------------------------------+
SELECT to_date('2009-07-30 04:17:52') >= to_date('2009-08-01 04:17:52');
+-------------------------------------------------------------+
|(to_date(2009-07-30 04:17:52) >= to_date(2009-08-01 04:17:52))|
+-------------------------------------------------------------+
|                                                        false|
+-------------------------------------------------------------+
SELECT 1 >= NULL;
+-----------+
|(1 >= NULL)|
+-----------+
|       NULL|
+-----------+
-- and
SELECT true and true;
+---------------+
|(true AND true)|
+---------------+
|           true|
+---------------+
SELECT true and false;
+----------------+
|(true AND false)|
+----------------+
|           false|
+----------------+
SELECT true and NULL;
+---------------+
|(true AND NULL)|
+---------------+
|           NULL|
+---------------+
SELECT false and NULL;
+----------------+
|(false AND NULL)|
+----------------+
|           false|
+----------------+
-- ilike
SELECT ilike('Wagon', '_Agon');
+-------------------+
|ilike(Wagon, _Agon)|
+-------------------+
|               true|
```

```
+-------------------+
SELECT '%SystemDrive%\Users\John' ilike '\%SystemDrive\%\\users%';
+-----------------------------------------------------------+
|ilike(%SystemDrive%\Users\John, \%SystemDrive\%\\users%)|
+-----------------------------------------------------------+
|                                                       true|
+-----------------------------------------------------------+
SELECT '%SystemDrive%\\USERS\\John' ilike '\%SystemDrive\%\\\\Users%';
+-----------------------------------------------------------+
|ilike(%SystemDrive%\USERS\John, \%SystemDrive\%\\Users%)|
+-----------------------------------------------------------+
|                                                       true|
+-----------------------------------------------------------+
SELECT '%SystemDrive%/Users/John' ilike '/%SYSTEMDrive/%//Users%' ESCAPE '/';
+-----------------------------------------------------------+
|ilike(%SystemDrive%/Users/John, /%SYSTEMDrive/%//Users%)|
+-----------------------------------------------------------+
|                                                       true|
+-----------------------------------------------------------+
-- in
SELECT 1 in(1, 2, 3);
+----------------+
|(1 IN (1, 2, 3))|
+----------------+
|            true|
+----------------+
SELECT 1 in(2, 3, 4);
+----------------+
|(1 IN (2, 3, 4))|
+----------------+
|           false|
+----------------+
SELECT named_struct('a', 1, 'b', 2) in(named_struct('a', 1, 'b', 1), named_struct('a',
 1, 'b', 3));
+-------------------------------------------------------------------------------------+
|(named_struct(a, 1, b, 2) IN (named_struct(a, 1, b, 1), named_struct(a, 1, b, 3)))|
+-------------------------------------------------------------------------------------+
|                                                                                false|
+-------------------------------------------------------------------------------------+
SELECT named_struct('a', 1, 'b', 2) in(named_struct('a', 1, 'b', 2), named_struct('a',
 1, 'b', 3));
+-------------------------------------------------------------------------------------+
|(named_struct(a, 1, b, 2) IN (named_struct(a, 1, b, 2), named_struct(a, 1, b, 3)))|
+-------------------------------------------------------------------------------------+
```

```
|                                                                                                  true|
+------------------------------------------------------------------------------------------------------+
-- isnan
SELECT isnan(cast('NaN' as double));
+--------------------------+
|isnan(CAST(NaN AS DOUBLE))|
+--------------------------+
|                      true|
+--------------------------+
-- isnotnull
SELECT isnotnull(1);
+---------------+
|(1 IS NOT NULL)|
+---------------+
|           true|
+---------------+
-- isnull
SELECT isnull(1);
+-----------+
|(1 IS NULL)|
+-----------+
|      false|
+-----------+
-- like
SELECT like('Wagon', '_Agon');
+----------------+
|Wagon LIKE _Agon|
+----------------+
|            true|
+----------------+
-- not
SELECT not true;
+----------+
|(NOT true)|
+----------+
|     false|
+----------+
SELECT not false;
+-----------+
|(NOT false)|
+-----------+
|       true|
+-----------+
SELECT not NULL;
```

```
+----------+
|(NOT NULL)|
+----------+
|      NULL|
+----------+
-- or
SELECT true or false;
+---------------+
|(true OR false)|
+---------------+
|           true|
+---------------+
SELECT false or false;
+----------------+
|(false OR false)|
+----------------+
|           false|
+----------------+
SELECT true or NULL;
+--------------+
|(true OR NULL)|
+--------------+
|          true|
+--------------+
SELECT false or NULL;
+---------------+
|(false OR NULL)|
+---------------+
|           NULL|
+---------------+
```

Funciones de mapa

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando SQL, consulte the section called “Comandos SQL compatibles”.

Función	Descripción
element_at(array, index)	Devuelve el elemento de la matriz en un índice dado (basado en 1).
element_at(map, key)	Devuelve el valor de una clave dada. La función devuelve NULL si la clave no se encuentra en el mapa.
map(key0, value0, key1, value1, ...)	Crea un mapa con los pares clave/valor dados.
map_concat(map, ...)	Devuelve la unión de todos los mapas dados
map_contains_key(map, key)	Devuelve true si el mapa contiene la clave especificada.
map_entries(map)	Devuelve una matriz desordenada de todas las entradas del mapa dado.
map_from_arrays(keys, values)	Crea un mapa con un par de las matrices clave/valor dadas. Todos los elementos de las claves no deben ser un valor nulo
map_from_entries(arrayOfEntries)	Devuelve un mapa creado a partir de la matriz de entradas dada.
map_keys(map)	Devuelve una matriz desordenada que contiene las claves del mapa.
map_values(map)	Devuelve una matriz desordenada que contiene los valores del mapa.
str_to_map(text[, pairDelim[, keyValueDelim]])	Crea un mapa después de dividir el texto en pares clave/valor mediante delimitadores. Los delimitadores predeterminados son ',' para `pairDelim` y ':' para `keyValueDelim`. Tanto `pairDelim` como `keyValueDelim` se tratan como expresiones regulares.

Función	Descripción
try_element_at(array, index)	Devuelve el elemento de la matriz en un índice dado (basado en 1). Si el índice es 0, el sistema generará un error. Si el índice es inferior a 0, accede a los elementos desde el último hasta el primero. La función siempre devuelve NULL si el índice supera la longitud de la matriz.
try_element_at(map, key)	Devuelve el valor de una clave dada. La función siempre devuelve NULL si la clave no está incluida en el mapa.

Ejemplos

```
-- element_at
SELECT element_at(array(1, 2, 3), 2);
+----------------------------+
|element_at(array(1, 2, 3), 2)|
+----------------------------+
|                           2|
+----------------------------+
SELECT element_at(map(1, 'a', 2, 'b'), 2);
+-----------------------------+
|element_at(map(1, a, 2, b), 2)|
+-----------------------------+
|                            b|
+-----------------------------+
-- map
SELECT map(1.0, '2', 3.0, '4');
+--------------------+
| map(1.0, 2, 3.0, 4)|
+--------------------+
|{1.0 -> 2, 3.0 -> 4}|
+--------------------+
-- map_concat
SELECT map_concat(map(1, 'a', 2, 'b'), map(3, 'c'));
+------------------------------------+
|map_concat(map(1, a, 2, b), map(3, c))|
```

```
+--------------------------------------+
|                  {1 -> a, 2 -> b, ...|
+--------------------------------------+
-- map_contains_key
SELECT map_contains_key(map(1, 'a', 2, 'b'), 1);
+-----------------------------------+
|map_contains_key(map(1, a, 2, b), 1)|
+-----------------------------------+
|                               true|
+-----------------------------------+
SELECT map_contains_key(map(1, 'a', 2, 'b'), 3);
+-----------------------------------+
|map_contains_key(map(1, a, 2, b), 3)|
+-----------------------------------+
|                              false|
+-----------------------------------+
-- map_entries
SELECT map_entries(map(1, 'a', 2, 'b'));
+----------------------------+
|map_entries(map(1, a, 2, b))|
+----------------------------+
|            [{1, a}, {2, b}]|
+----------------------------+
-- map_from_arrays
SELECT map_from_arrays(array(1.0, 3.0), array('2', '4'));
+-------------------------------------------+
|map_from_arrays(array(1.0, 3.0), array(2, 4))|
+-------------------------------------------+
|                        {1.0 -> 2, 3.0 -> 4}|
+-------------------------------------------+
-- map_from_entries
SELECT map_from_entries(array(struct(1, 'a'), struct(2, 'b')));
+---------------------------------------------------+
|map_from_entries(array(struct(1, a), struct(2, b)))|
+---------------------------------------------------+
|                                {1 -> a, 2 -> b}|
+---------------------------------------------------+
-- map_keys
SELECT map_keys(map(1, 'a', 2, 'b'));
+-------------------------+
|map_keys(map(1, a, 2, b))|
+-------------------------+
|                   [1, 2]|
+-------------------------+
```

```
-- map_values
SELECT map_values(map(1, 'a', 2, 'b'));
+--------------------------+
|map_values(map(1, a, 2, b))|
+--------------------------+
|                    [a, b]|
+--------------------------+
-- str_to_map
SELECT str_to_map('a:1,b:2,c:3', ',', ':');
+----------------------------+
|str_to_map(a:1,b:2,c:3, ,, :)|
+----------------------------+
|         {a -> 1, b -> 2, ...|
+----------------------------+
SELECT str_to_map('a');
+------------------+
|str_to_map(a, ,, :)|
+------------------+
|        {a -> NULL}|
+------------------+
-- try_element_at
SELECT try_element_at(array(1, 2, 3), 2);
+---------------------------------+
|try_element_at(array(1, 2, 3), 2)|
+---------------------------------+
|                                2|
+---------------------------------+
SELECT try_element_at(map(1, 'a', 2, 'b'), 2);
+---------------------------------+
|try_element_at(map(1, a, 2, b), 2)|
+---------------------------------+
|                                 b|
+---------------------------------+
```

Funciones matemáticas

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando SQL, consulte the section called “Comandos SQL compatibles”.

Función	Descripción
expr1 % expr2	Devuelve el resto después de `expr1`/`expr2`.
expr1 * expr2	Devuelve `expr1` por `expr2`.
expr1 + expr2	Devuelve `expr1` más `expr2`.
expr1 - expr2	Devuelve `expr1` menos `expr2`.
expr1 / expr2	Devuelve `expr1` dividido `expr2`. Siempre realiza una división de punto flotante.
abs(expr)	Devuelve el valor absoluto del valor numérico o del intervalo.
acos(expr)	Devuelve el coseno inverso (también conocido como arcoseno) de `expr`, como si se hubiera calculado con `java.lang.math.acos`.
acosh(expr)	Devuelve el coseno hiperbólico inverso de `expr`.
asin(expr)	Devuelve el seno inverso (también conocido como arco seno), el arco seno de `expr`, como si se hubiera calculado con `java.lang.math.asin`.
asinh(expr)	Devuelve el seno hiperbólico inverso de `expr`.
atan(expr)	Devuelve la tangente inversa (también conocida como arco tangente) de `expr`, como si la hubiera calculado `java.lang.math.atan`
atan2(exprY, exprX)	Devuelve el ángulo en radianes entre el eje x positivo de un plano y el punto dado por las coordenadas (`exprX`, `exprY`), como si se hubiera calculado con `java.lang.Math.atan2`.

Función	Descripción
atanh(expr)	Devuelve la tangente hiperbólica inversa de `expr`.
bin(expr)	Devuelve la representación en cadena del valor largo `expr` representado en binario.
bround(expr, d)	Devuelve `expr` redondeado a `d` decimales utilizando el modo de redondeo HALF_EVEN.
cbrt(expr)	Devuelve la raíz cúbica de `expr`.
ceil(expr[, scale])	Devuelve el número más pequeño después de redondear hacia arriba que no sea menor que `expr`. Se puede especificar un parámetro `scale` opcional para controlar el comportam iento de redondeo.
ceiling(expr[, scale])	Devuelve el número más pequeño después de redondear hacia arriba que no sea menor que `expr`. Se puede especificar un parámetro `scale` opcional para controlar el comportam iento de redondeo.
conv(num, from_base, to_base)	Convierte `num` de `from_base` a `to_base`.
cos(expr)	Devuelve el coseno de `expr`, como si lo hubiera calculado `java.lang.Math.cos`.
cosh(expr)	Devuelve el coseno hiperbólico de `expr`, como si se hubiera calculado con `java.lan g.Math.cosh`.
cot(expr)	Devuelve la cotangente de `expr`, como si se hubiera calculado con `1/java.lang.Math.tan`.
csc(expr)	Devuelve la cosecante de `expr`, como si se hubiera calculado con `1/java.lang.Math.sin`.

Función	Descripción
degrees(expr)	Convierte radianes en grados.
expr1 div expr2	Divide `expr1` por `expr2`. Devuelve NULL si un operando es NULL o `expr2` es 0. El resultado se convierte en largo.
e()	Devuelve el número de Euler, e.
expr(expr)	Devuelve e a la potencia de `expr`.
expm1(expr) - Devuelve exp(`expr`)	1
factorial(expr)	Devuelve la factorial de `expr`. `expr` es [0.. 20]. De lo contrario, es nulo.
floor(expr[, scale])	Devuelve el número mayor después de redondear para abajo que no sea mayor que `expr`. Se puede especificar un parámetro `scale` opcional para controlar el comportam iento de redondeo.
greatest(expr, ...)	Devuelve el valor máximo de todos los parámetros, omitiendo los valores nulos.
hex(expr)	Convierte `expr` en hexadecimal.
hypot(expr1, expr2)	Devuelve sqrt(`expr1`**2 + `expr2`**2).
least(expr, ...)	Devuelve el valor mínimo de todos los parámetros, omitiendo los valores nulos.
ln(expr)	Devuelve el logaritmo natural (de base e) de `expr`.
log(base, expr)	Devuelve el logaritmo de `expr` con `base`.
log10(expr)	Devuelve el logaritmo en base 10 de `expr`.

Función	Descripción
log1p(expr)	Devuelve log(1 + `expr`).
log2(expr)	Devuelve el logaritmo de `expr` con base 2.
expr1 mod expr2	Devuelve el resto después de `expr1`/`expr2`.
negative(expr)	Devuelve el valor negado de `expr`.
pi()	Devuelve pi.
pmod(expr1, expr2)	Devuelve el valor positivo del resto de `expr1`/`expr2`.
positive(expr)	Devuelve el valor de `expr`.
pow(expr1, expr2)	Eleva `expr1` a la potencia de `expr2`.
power(expr1, expr2)	Eleva `expr1` a la potencia de `expr2`.
radians(expr)	Convierte grados en radianes.
rand([seed])	Devuelve un valor aleatorio con valores independientes e idénticos (i.i.d.) distribuidos uniformemente en [0, 1).
randn([seed])	Devuelve un valor aleatorio con valores independientes y de distribución idéntica (i.i.d.) extraídos de la distribución normal estándar.
random([seed])	Devuelve un valor aleatorio con valores independientes e idénticos (i.i.d.) distribuidos uniformemente en [0, 1).
rint(expr)	Devuelve el valor doble que tenga el valor más cercano al argumento y que sea igual a un entero matemático.

Función	Descripción
round(expr, d)	Devuelve `expr` redondeado a `d` decimales utilizando el modo de redondeo HALF_UP.
sec(expr)	Devuelve la secante de `expr`, como si se hubiera calculado con `1/java.lang.Math.cos`.
shiftleft(base, expr)	Cambio bit a bit hacia la izquierda.
sign(expr)	Devuelve -1.0, 0.0 o 1.0 si `expr` es negativo, 0 o positivo.
signum(expr)	Devuelve -1.0, 0.0 o 1.0 si `expr` es negativo, 0 o positivo.
sin(expr)	Devuelve el seno de `expr`, como si lo hubiera calculado `java.lang.Math.sin`.
sinh(expr)	Devuelve el seno hiperbólico de `expr`, como si se hubiera calculado mediante `java.lang.Math.sinh`.
sqrt(expr)	Devuelve la raíz cuadrada de `expr`.
tan(expr)	Devuelve la tangente de `expr`, como si se hubiera calculado con `java.lang.Math.tan`.
tanh(expr)	Devuelve la tangente hiperbólica de `expr`, como si se hubiera calculado con `java.lang.Math.tanh`.
try_add(expr1, expr2)	Devuelve la suma de `expr1`y `expr2` y el resultado es nulo en caso de desbordamiento. Los tipos de entrada aceptables son los mismos que con el operador `+`.

Función	Descripción
try_divide(dividend, divisor)	Devuelve `dividend`/`divisor`. Siempre realiza una división de punto flotante. Su resultado siempre es nulo si `expr2` es 0. El `dividend` debe ser un número o un intervalo. El `divisor` debe ser numérico.
try_multiply(expr1, expr2)	Devuelve `expr1`*`expr2` y el resultado es nulo en caso de desbordamiento. Los tipos de entrada aceptables son los mismos que con el operador `*`.
try_subtract(expr1, expr2)	Devuelve `expr1`-`expr2` y el resultado es nulo en caso de desbordamiento. Los tipos de entrada aceptables son los mismos para el operador `-`.
unhex(expr)	Convierte `expr` hexadecimal en binario.
width_bucket(value, min_value, max_value, num_bucket)	Devuelve el número de bucket al que se asignaría `value` en un histograma de ancho equivalente con buckets de `num_bucket`, en el rango de `min_value` a `max_value`.

Ejemplos

```
-- %
SELECT 2 % 1.8;
+---------+
|(2 % 1.8)|
+---------+
|      0.2|
+---------+
SELECT MOD(2, 1.8);
+-----------+
|mod(2, 1.8)|
+-----------+
|        0.2|
```

```
+-----------+
-- *
SELECT 2 * 3;
+-------+
|(2 * 3)|
+-------+
|      6|
+-------+
-- +
SELECT 1 + 2;
+-------+
|(1 + 2)|
+-------+
|      3|
+-------+
-- -
SELECT 2 - 1;
+-------+
|(2 - 1)|
+-------+
|      1|
+-------+
-- /
SELECT 3 / 2;
+-------+
|(3 / 2)|
+-------+
|    1.5|
+-------+
SELECT 2L / 2L;
+-------+
|(2 / 2)|
+-------+
|    1.0|
+-------+
-- abs
SELECT abs(-1);
+-------+
|abs(-1)|
+-------+
|      1|
+-------+
SELECT abs(INTERVAL -'1-1' YEAR TO MONTH);
+----------------------------------+
```

```
|abs(INTERVAL '-1-1' YEAR TO MONTH)|
+----------------------------------+
|              INTERVAL '1-1' YE...|
+----------------------------------+
-- acos
SELECT acos(1);
+-------+
|ACOS(1)|
+-------+
|    0.0|
+-------+
SELECT acos(2);
+-------+
|ACOS(2)|
+-------+
|    NaN|
+-------+
-- acosh
SELECT acosh(1);
+--------+
|ACOSH(1)|
+--------+
|     0.0|
+--------+
SELECT acosh(0);
+--------+
|ACOSH(0)|
+--------+
|     NaN|
+--------+
-- asin
SELECT asin(0);
+-------+
|ASIN(0)|
+-------+
|    0.0|
+-------+
SELECT asin(2);
+-------+
|ASIN(2)|
+-------+
|    NaN|
+-------+
-- asinh
```

```
SELECT asinh(0);
+--------+
|ASINH(0)|
+--------+
|     0.0|
+--------+
-- atan
SELECT atan(0);
+-------+
|ATAN(0)|
+-------+
|    0.0|
+-------+
-- atan2
SELECT atan2(0, 0);
+-----------+
|ATAN2(0, 0)|
+-----------+
|        0.0|
+-----------+
-- atanh
SELECT atanh(0);
+--------+
|ATANH(0)|
+--------+
|     0.0|
+--------+
SELECT atanh(2);
+--------+
|ATANH(2)|
+--------+
|     NaN|
+--------+
-- bin
SELECT bin(13);
+-------+
|bin(13)|
+-------+
|   1101|
+-------+
SELECT bin(-13);
+--------------------+
|            bin(-13)|
+--------------------+
```

```
|1111111111111111...|
+--------------------+
SELECT bin(13.3);
+---------+
|bin(13.3)|
+---------+
|     1101|
+---------+
-- bround
SELECT bround(2.5, 0);
+--------------+
|bround(2.5, 0)|
+--------------+
|             2|
+--------------+
SELECT bround(25, -1);
+--------------+
|bround(25, -1)|
+--------------+
|            20|
+--------------+
-- cbrt
SELECT cbrt(27.0);
+----------+
|CBRT(27.0)|
+----------+
|       3.0|
+----------+
-- ceil
SELECT ceil(-0.1);
+----------+
|CEIL(-0.1)|
+----------+
|         0|
+----------+
SELECT ceil(5);
+-------+
|CEIL(5)|
+-------+
|      5|
+-------+
SELECT ceil(3.1411, 3);
+---------------+
|ceil(3.1411, 3)|
```

```
+---------------+
|          3.142|
+---------------+
SELECT ceil(3.1411, -3);
+----------------+
|ceil(3.1411, -3)|
+----------------+
|            1000|
+----------------+
-- ceiling
SELECT ceiling(-0.1);
+-------------+
|ceiling(-0.1)|
+-------------+
|            0|
+-------------+
SELECT ceiling(5);
+----------+
|ceiling(5)|
+----------+
|         5|
+----------+
SELECT ceiling(3.1411, 3);
+------------------+
|ceiling(3.1411, 3)|
+------------------+
|             3.142|
+------------------+
SELECT ceiling(3.1411, -3);
+-------------------+
|ceiling(3.1411, -3)|
+-------------------+
|               1000|
+-------------------+
-- conv
SELECT conv('100', 2, 10);
+----------------+
|conv(100, 2, 10)|
+----------------+
|               4|
+----------------+
SELECT conv(-10, 16, -10);
+------------------+
|conv(-10, 16, -10)|
```

```
+------------------+
|               -16|
+------------------+
-- cos
SELECT cos(0);
+------+
|COS(0)|
+------+
|   1.0|
+------+
-- cosh
SELECT cosh(0);
+-------+
|COSH(0)|
+-------+
|    1.0|
+-------+
-- cot
SELECT cot(1);
+------------------+
|            COT(1)|
+------------------+
|0.6420926159343306|
+------------------+
-- csc
SELECT csc(1);
+------------------+
|            CSC(1)|
+------------------+
|1.1883951057781212|
+------------------+
-- degrees
SELECT degrees(3.141592653589793);
+--------------------------+
|DEGREES(3.141592653589793)|
+--------------------------+
|                     180.0|
+--------------------------+
-- div
SELECT 3 div 2;
+---------+
|(3 div 2)|
+---------+
|        1|
```

```
+---------+
SELECT INTERVAL '1-1' YEAR TO MONTH div INTERVAL '-1' MONTH;
+-----------------------------------------------------------+
|(INTERVAL '1-1' YEAR TO MONTH div INTERVAL '-1' MONTH)|
+-----------------------------------------------------------+
|                                                        -13|
+-----------------------------------------------------------+
-- e
SELECT e();
+-----------------+
|              E()|
+-----------------+
|2.718281828459045|
+-----------------+
-- exp
SELECT exp(0);
+------+
|EXP(0)|
+------+
|   1.0|
+------+
-- expm1
SELECT expm1(0);
+--------+
|EXPM1(0)|
+--------+
|     0.0|
+--------+
-- factorial
SELECT factorial(5);
+------------+
|factorial(5)|
+------------+
|         120|
+------------+
-- floor
SELECT floor(-0.1);
+-----------+
|FLOOR(-0.1)|
+-----------+
|         -1|
+-----------+
SELECT floor(5);
+--------+
```

```
|FLOOR(5)|
+--------+
|       5|
+--------+
SELECT floor(3.1411, 3);
+----------------+
|floor(3.1411, 3)|
+----------------+
|           3.141|
+----------------+
SELECT floor(3.1411, -3);
+-----------------+
|floor(3.1411, -3)|
+-----------------+
|                0|
+-----------------+
-- greatest
SELECT greatest(10, 9, 2, 4, 3);
+------------------------+
|greatest(10, 9, 2, 4, 3)|
+------------------------+
|                      10|
+------------------------+
-- hex
SELECT hex(17);
+-------+
|hex(17)|
+-------+
|     11|
+-------+
SELECT hex('SQL');
+------------------+
|    hex(SQL)|
+------------------+
|53514C|
+------------------+
-- hypot
SELECT hypot(3, 4);
+-----------+
|HYPOT(3, 4)|
+-----------+
|        5.0|
+-----------+
-- least
```

```
SELECT least(10, 9, 2, 4, 3);
+---------------------+
|least(10, 9, 2, 4, 3)|
+---------------------+
|                    2|
+---------------------+
-- ln
SELECT ln(1);
+-----+
|ln(1)|
+-----+
|  0.0|
+-----+
-- log
SELECT log(10, 100);
+------------+
|LOG(10, 100)|
+------------+
|         2.0|
+------------+
-- log10
SELECT log10(10);
+---------+
|LOG10(10)|
+---------+
|      1.0|
+---------+
-- log1p
SELECT log1p(0);
+--------+
|LOG1P(0)|
+--------+
|     0.0|
+--------+
-- log2
SELECT log2(2);
+-------+
|LOG2(2)|
+-------+
|    1.0|
+-------+
-- mod
SELECT 2 % 1.8;
+---------+
```

```
|(2 % 1.8)|
+---------+
|      0.2|
+---------+
SELECT MOD(2, 1.8);
+-----------+
|mod(2, 1.8)|
+-----------+
|        0.2|
+-----------+
-- negative
SELECT negative(1);
+-----------+
|negative(1)|
+-----------+
|         -1|
+-----------+
-- pi
SELECT pi();
+-----------------+
|             PI()|
+-----------------+
|3.141592653589793|
+-----------------+
-- pmod
SELECT pmod(10, 3);
+-----------+
|pmod(10, 3)|
+-----------+
|          1|
+-----------+
SELECT pmod(-10, 3);
+------------+
|pmod(-10, 3)|
+------------+
|           2|
+------------+
-- positive
SELECT positive(1);
+-----+
|(+ 1)|
+-----+
|    1|
+-----+
```

```
-- pow
SELECT pow(2, 3);
+---------+
|pow(2, 3)|
+---------+
|      8.0|
+---------+
-- power
SELECT power(2, 3);
+-----------+
|POWER(2, 3)|
+-----------+
|        8.0|
+-----------+
-- radians
SELECT radians(180);
+-----------------+
|     RADIANS(180)|
+-----------------+
|3.141592653589793|
+-----------------+
-- rand
SELECT rand();
+------------------+
|            rand()|
+------------------+
|0.7211420708112387|
+------------------+
SELECT rand(0);
+------------------+
|           rand(0)|
+------------------+
|0.7604953758285915|
+------------------+
SELECT rand(null);
+------------------+
|        rand(NULL)|
+------------------+
|0.7604953758285915|
+------------------+
-- randn
SELECT randn();
+-------------------+
|             randn()|
```

```
+-------------------+
|-0.8175603217732732|
+-------------------+
SELECT randn(0);
+------------------+
|          randn(0)|
+------------------+
|1.6034991609278433|
+------------------+
SELECT randn(null);
+------------------+
|       randn(NULL)|
+------------------+
|1.6034991609278433|
+------------------+
-- random
SELECT random();
+-----------------+
|           rand()|
+-----------------+
|0.394205008255365|
+-----------------+
SELECT random(0);
+------------------+
|           rand(0)|
+------------------+
|0.7604953758285915|
+------------------+
SELECT random(null);
+------------------+
|        rand(NULL)|
+------------------+
|0.7604953758285915|
+------------------+
-- rint
SELECT rint(12.3456);
+-------------+
|rint(12.3456)|
+-------------+
|         12.0|
+-------------+
-- round
SELECT round(2.5, 0);
+-------------+
```

```
|round(2.5, 0)|
+-------------+
|            3|
+-------------+
-- sec
SELECT sec(0);
+------+
|SEC(0)|
+------+
|   1.0|
+------+
-- shiftleft
SELECT shiftleft(2, 1);
+---------------+
|shiftleft(2, 1)|
+---------------+
|              4|
+---------------+
-- sign
SELECT sign(40);
+--------+
|sign(40)|
+--------+
|     1.0|
+--------+
SELECT sign(INTERVAL -'100' YEAR);
+--------------------------+
|sign(INTERVAL '-100' YEAR)|
+--------------------------+
|                      -1.0|
+--------------------------+
-- signum
SELECT signum(40);
+----------+
|SIGNUM(40)|
+----------+
|       1.0|
+----------+
SELECT signum(INTERVAL -'100' YEAR);
+----------------------------+
|SIGNUM(INTERVAL '-100' YEAR)|
+----------------------------+
|                        -1.0|
+----------------------------+
```

```
-- sin
SELECT sin(0);
+------+
|SIN(0)|
+------+
|   0.0|
+------+
-- sinh
SELECT sinh(0);
+-------+
|SINH(0)|
+-------+
|    0.0|
+-------+
-- sqrt
SELECT sqrt(4);
+-------+
|SQRT(4)|
+-------+
|    2.0|
+-------+
-- tan
SELECT tan(0);
+------+
|TAN(0)|
+------+
|   0.0|
+------+
-- tanh
SELECT tanh(0);
+-------+
|TANH(0)|
+-------+
|    0.0|
+-------+
-- try_add
SELECT try_add(1, 2);
+-------------+
|try_add(1, 2)|
+-------------+
|            3|
+-------------+
SELECT try_add(2147483647, 1);
+----------------------+
```

```
|try_add(2147483647, 1)|
+----------------------+
|                  NULL|
+----------------------+
SELECT try_add(date'2021-01-01', 1);
+-----------------------------+
|try_add(DATE '2021-01-01', 1)|
+-----------------------------+
|                   2021-01-02|
+-----------------------------+
SELECT try_add(date'2021-01-01', interval 1 year);
+-------------------------------------------+
|try_add(DATE '2021-01-01', INTERVAL '1' YEAR)|
+-------------------------------------------+
|                                 2022-01-01|
+-------------------------------------------+
SELECT try_add(timestamp'2021-01-01 00:00:00', interval 1 day);
+-----------------------------------------------------------+
|try_add(TIMESTAMP '2021-01-01 00:00:00', INTERVAL '1' DAY)|
+-----------------------------------------------------------+
|                                      2021-01-02 00:00:00|
+-----------------------------------------------------------+
SELECT try_add(interval 1 year, interval 2 year);
+-----------------------------------------+
|try_add(INTERVAL '1' YEAR, INTERVAL '2' YEAR)|
+-----------------------------------------+
|                           INTERVAL '3' YEAR|
+-----------------------------------------+
-- try_divide
SELECT try_divide(3, 2);
+----------------+
|try_divide(3, 2)|
+----------------+
|             1.5|
+----------------+
SELECT try_divide(2L, 2L);
+----------------+
|try_divide(2, 2)|
+----------------+
|             1.0|
+----------------+
SELECT try_divide(1, 0);
+----------------+
|try_divide(1, 0)|
```

```
+----------------+
|            NULL|
+----------------+
SELECT try_divide(interval 2 month, 2);
+--------------------------------+
|try_divide(INTERVAL '2' MONTH, 2)|
+--------------------------------+
|             INTERVAL '0-1' YE...|
+--------------------------------+
SELECT try_divide(interval 2 month, 0);
+--------------------------------+
|try_divide(INTERVAL '2' MONTH, 0)|
+--------------------------------+
|                            NULL|
+--------------------------------+
-- try_multiply
SELECT try_multiply(2, 3);
+------------------+
|try_multiply(2, 3)|
+------------------+
|                 6|
+------------------+
SELECT try_multiply(-2147483648, 10);
+-----------------------------+
|try_multiply(-2147483648, 10)|
+-----------------------------+
|                         NULL|
+-----------------------------+
SELECT try_multiply(interval 2 year, 3);
+---------------------------------+
|try_multiply(INTERVAL '2' YEAR, 3)|
+---------------------------------+
|              INTERVAL '6-0' YE...|
+---------------------------------+
-- try_subtract
SELECT try_subtract(2, 1);
+------------------+
|try_subtract(2, 1)|
+------------------+
|                 1|
+------------------+
SELECT try_subtract(-2147483648, 1);
+----------------------------+
|try_subtract(-2147483648, 1)|
```

```
+---------------------------+
|                       NULL|
+---------------------------+
SELECT try_subtract(date'2021-01-02', 1);
+---------------------------------+
|try_subtract(DATE '2021-01-02', 1)|
+---------------------------------+
|                       2021-01-01|
+---------------------------------+
SELECT try_subtract(date'2021-01-01', interval 1 year);
+-------------------------------------------------+
|try_subtract(DATE '2021-01-01', INTERVAL '1' YEAR)|
+-------------------------------------------------+
|                                       2020-01-01|
+-------------------------------------------------+
SELECT try_subtract(timestamp'2021-01-02 00:00:00', interval 1 day);
+-----------------------------------------------------------------+
|try_subtract(TIMESTAMP '2021-01-02 00:00:00', INTERVAL '1' DAY)|
+-----------------------------------------------------------------+
|                                              2021-01-01 00:00:00|
+-----------------------------------------------------------------+
SELECT try_subtract(interval 2 year, interval 1 year);
+-------------------------------------------------+
|try_subtract(INTERVAL '2' YEAR, INTERVAL '1' YEAR)|
+-------------------------------------------------+
|                                 INTERVAL '1' YEAR|
+-------------------------------------------------+
-- unhex
SELECT decode(unhex('53514C'), 'UTF-8');
+---------------------------------------+
|decode(unhex(53514C), UTF-8)|
+---------------------------------------+
|                                  SQL|
+---------------------------------------+
-- width_bucket
SELECT width_bucket(5.3, 0.2, 10.6, 5);
+-------------------------------+
|width_bucket(5.3, 0.2, 10.6, 5)|
+-------------------------------+
|                              3|
+-------------------------------+
SELECT width_bucket(-2.1, 1.3, 3.4, 3);
+-------------------------------+
|width_bucket(-2.1, 1.3, 3.4, 3)|
```

```
+------------------------------+
|                             0|
+------------------------------+
SELECT width_bucket(8.1, 0.0, 5.7, 4);
+------------------------------+
|width_bucket(8.1, 0.0, 5.7, 4)|
+------------------------------+
|                             5|
+------------------------------+
SELECT width_bucket(-0.9, 5.2, 0.5, 2);
+-------------------------------+
|width_bucket(-0.9, 5.2, 0.5, 2)|
+-------------------------------+
|                              3|
+-------------------------------+
SELECT width_bucket(INTERVAL '0' YEAR, INTERVAL '0' YEAR, INTERVAL '10' YEAR, 10);
+-----------------------------------------------------------------------------+
|width_bucket(INTERVAL '0' YEAR, INTERVAL '0' YEAR, INTERVAL '10' YEAR, 10)|
+-----------------------------------------------------------------------------+
|                                                                            1|
+-----------------------------------------------------------------------------+
SELECT width_bucket(INTERVAL '1' YEAR, INTERVAL '0' YEAR, INTERVAL '10' YEAR, 10);
+-----------------------------------------------------------------------------+
|width_bucket(INTERVAL '1' YEAR, INTERVAL '0' YEAR, INTERVAL '10' YEAR, 10)|
+-----------------------------------------------------------------------------+
|                                                                            2|
+-----------------------------------------------------------------------------+
SELECT width_bucket(INTERVAL '0' DAY, INTERVAL '0' DAY, INTERVAL '10' DAY, 10);
+---------------------------------------------------------------------+
|width_bucket(INTERVAL '0' DAY, INTERVAL '0' DAY, INTERVAL '10' DAY, 10)|
+---------------------------------------------------------------------+
|                                                                    1|
+---------------------------------------------------------------------+
SELECT width_bucket(INTERVAL '1' DAY, INTERVAL '0' DAY, INTERVAL '10' DAY, 10);
+---------------------------------------------------------------------+
|width_bucket(INTERVAL '1' DAY, INTERVAL '0' DAY, INTERVAL '10' DAY, 10)|
+---------------------------------------------------------------------+
|                                                                    2|
+---------------------------------------------------------------------+
```

Funciones generadoras

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten estas funciones SQL, consulte the section called “Comandos SQL compatibles”.

Función	Descripción
explode(expr)	Separa los elementos de la matriz `expr` en varias filas o los elementos del mapa `expr` en varias filas y columnas. A menos que se especifique lo contrario, utiliza el nombre de columna predeterminado `col` para los elementos de la matriz o `key` y `value` para los elementos del mapa.
explode_outer(expr)	Separa los elementos de la matriz `expr` en varias filas o los elementos del mapa `expr` en varias filas y columnas. A menos que se especifique lo contrario, utiliza el nombre de columna predeterminado `col` para los elementos de la matriz o `key` y `value` para los elementos del mapa.
inline(expr)	Descompone un conjunto de estructuras en una tabla. Utiliza los nombres de columna col1, col2, etc. de forma predeterminada, a menos que se especifique lo contrario.
inline_outer(expr)	Descompone un conjunto de estructuras en una tabla. Utiliza los nombres de columna col1, col2, etc. de forma predeterminada, a menos que se especifique lo contrario.
posexplode(expr)	Separa los elementos de la matriz `expr` en varias filas con posicione s, o los elementos del mapa `expr` en varias filas y columnas con posiciones. A menos que se especifique lo contrario, utiliza el nombre de columna `pos` para la posición, `col` para los elementos de la matriz o `key` y `value` para los elementos del mapa.
posexplode_outer(e xpr)	Separa los elementos de la matriz `expr` en varias filas con posicione s, o los elementos del mapa `expr` en varias filas y columnas con posiciones. A menos que se especifique lo contrario, utiliza el nombre

Función	Descripción
	de columna `pos` para la posición, `col` para los elementos de la matriz o `key` y `value` para los elementos del mapa.
stack(n, expr1, ..., exprk)	Separa `expr1`,..., `expr` en `n` filas. Utiliza los nombres de columna col0, col1, etc. de forma predeterminada a menos que se especifique lo contrario.

Ejemplos

```
-- explode
SELECT explode(array(10, 20));
+---+
|col|
+---+
| 10|
| 20|
+---+

SELECT explode(collection => array(10, 20));
+---+
|col|
+---+
| 10|
| 20|
+---+

SELECT * FROM explode(collection => array(10, 20));
+---+
|col|
+---+
| 10|
| 20|
+---+

-- explode_outer
SELECT explode_outer(array(10, 20));
+---+
|col|
+---+
| 10|
```

```
| 20|
+---+

SELECT explode_outer(collection => array(10, 20));
+---+
|col|
+---+
| 10|
| 20|
+---+

SELECT * FROM explode_outer(collection => array(10, 20));
+---+
|col|
+---+
| 10|
| 20|
+---+

-- inline
SELECT inline(array(struct(1, 'a'), struct(2, 'b')));
+----+----+
|col1|col2|
+----+----+
|   1|   a|
|   2|   b|
+----+----+

-- inline_outer
SELECT inline_outer(array(struct(1, 'a'), struct(2, 'b')));
+----+----+
|col1|col2|
+----+----+
|   1|   a|
|   2|   b|
+----+----+

-- posexplode
SELECT posexplode(array(10,20));
+---+---+
|pos|col|
+---+---+
|  0| 10|
|  1| 20|
```

```
+---+---+

SELECT * FROM posexplode(array(10,20));
+---+---+
|pos|col|
+---+---+
|  0| 10|
|  1| 20|
+---+---+

-- posexplode_outer
SELECT posexplode_outer(array(10,20));
+---+---+
|pos|col|
+---+---+
|  0| 10|
|  1| 20|
+---+---+

SELECT * FROM posexplode_outer(array(10,20));
+---+---+
|pos|col|
+---+---+
|  0| 10|
|  1| 20|
+---+---+

-- stack
SELECT stack(2, 1, 2, 3);
+----+----+
|col0|col1|
+----+----+
|   1|   2|
|   3|NULL|
+----+----+
```

Cláusula SELECT

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando SQL, consulte the section called “Comandos SQL compatibles”.

OpenSearch SQL admite una sentencia `SELECT` utilizada para recuperar conjuntos de resultados de una o más tablas. En la siguiente sección se describe la sintaxis general de la consulta y las diferentes estructuras de una consulta.

Sintaxis

```
select_statement
[ { UNION | INTERSECT | EXCEPT } [ ALL | DISTINCT ] select_statement, ... ]
[ ORDER BY
    { expression [ ASC | DESC ] [ NULLS { FIRST | LAST } ]
    [ , ... ]
    }
]
[ SORT BY
    { expression [ ASC | DESC ] [ NULLS { FIRST | LAST } ]
    [ , ... ]
    }
]
[ WINDOW { named_window [ , WINDOW named_window, ... ] } ]
[ LIMIT { ALL | expression } ]
```

Mientras que `select_statement` se define como:

```
SELECT [ ALL | DISTINCT ] { [ [ named_expression ] [ , ... ] ] }
FROM { from_item [ , ... ] }
[ PIVOT clause ]
[ UNPIVOT clause ]
[ LATERAL VIEW clause ] [ ... ]
[ WHERE boolean_expression ]
[ GROUP BY expression [ , ... ] ]
[ HAVING boolean_expression ]
```

Parámetros

- ALL

  Selecciona todas las filas coincidentes de la relación y está habilitada de forma predeterminada.
- DISTINCT

  Selecciona todas las filas coincidentes de la relación después de eliminar los duplicados de los resultados.

- named_expression

  Una expresión con un nombre asignado. En general, denota una expresión de columna.

  Sintaxis: `expression [[AS] alias]`

- from_item

  Relación de tabla

  Relación de unión

  Relación de tablas dinámicas

  Relación no dinámica

  Funciones con valores de tabla

  Tabla en línea

  `[ LATERAL ] ( Subquery )`

- PIVOT

  La cláusula `PIVOT` se utiliza desde la perspectiva de los datos. Puede obtener los valores agregados en función de un valor de columna específico.

- UNPIVOT

  La cláusula `UNPIVOT` transforma las columnas en filas. Es lo contrario a `PIVOT`, excepto en lo que respecta a la agregación de valores.

- LATERAL VIEW

  La cláusula `LATERAL VIEW` se usa junto con funciones generadoras, por ejemplo `EXPLODE`, que generarán una tabla virtual que contiene una o más filas.

  `LATERAL VIEW` aplicará las filas a cada fila de salida original.

- WHERE

  Filtra el resultado de la cláusula `FROM` en función de los predicados proporcionados.

- GROUP BY

  Especifica las expresiones que se utilizan para agrupar las filas.

Se utiliza junto con las funciones de agregado (`MIN`, `MAX`, `COUNT`, `SUM`, `AVG`, etc.) para agrupar las filas en función de las expresiones de agrupación y los valores agregados de cada grupo.

Cuando se adjunta una cláusula `FILTER` a una función agregada, solo las filas coincidentes se pasan a esa función.

- HAVING

  Especifica los predicados mediante los que se filtran las filas producidas por `GROUP BY`.

  La cláusula `HAVING` se utiliza para filtrar las filas una vez realizada la agrupación.

  Si `HAVING` se especifica sin `GROUP BY`, indica que `GROUP BY` no tiene expresiones de agrupamiento (agregado global).

- ORDER BY

  Especifica el orden de las filas del conjunto completo de resultados de la consulta.

  Las filas de salida se ordenan en las particiones.

  Este parámetro se excluye mutuamente con `SORT BY` y `DISTRIBUTE BY` no se puede especificar de forma conjunta.

- SORT BY

  Especifica el orden por el que se ordenan las filas dentro de cada partición.

  Este parámetro se excluye mutuamente con `ORDER BY` y no se puede especificar de forma conjunta.

- LIMIT

  Especifica el número máximo de filas que puede devolver una sentencia o subconsulta.

  Esta cláusula se usa principalmente junto con `ORDER BY` para producir un resultado determinista.

- boolean_expression

  Especifica cualquier expresión que se evalúe como un valor booleano.

  Se pueden combinar dos o más expresiones mediante los operadores lógicos (`AND`, `OR`).

- expresión

Combinación de uno o varios valores, operadores o funciones SQL que dan como resultado un valor.

- named_window

  Especifica los alias de una o más especificaciones de la ventana de origen.

  Se puede hacer referencia a las especificaciones de la ventana de origen en las definiciones de ventana de la consulta.

Cláusula WHERE

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando SQL, consulte the section called “Comandos SQL compatibles”.

La cláusula `WHERE` se utiliza para limitar los resultados de la cláusula `FROM` de una consulta o subconsulta en función de la condición especificada.

Sintaxis

```
WHERE boolean_expression
```

Parámetros

- boolean_expression

  Especifica cualquier expresión que se evalúe como un valor booleano.

  Se pueden combinar dos o más expresiones mediante los operadores lógicos (`AND`, `OR`).

Ejemplos

```
CREATE TABLE person (id INT, name STRING, age INT);
INSERT INTO person VALUES
(100, 'John', 30),
(200, 'Mary', NULL),
(300, 'Mike', 80),
```

```
(400, 'Dan',  50);

-- Comparison operator in `WHERE` clause.
SELECT * FROM person WHERE id > 200 ORDER BY id;
+---+----+---+
| id|name|age|
+---+----+---+
|300|Mike| 80|
|400| Dan| 50|
+---+----+---+

-- Comparison and logical operators in `WHERE` clause.
SELECT * FROM person WHERE id = 200 OR id = 300 ORDER BY id;
+---+----+----+
| id|name| age|
+---+----+----+
|200|Mary|null|
|300|Mike|  80|
+---+----+----+

-- IS NULL expression in `WHERE` clause.
SELECT * FROM person WHERE id > 300 OR age IS NULL ORDER BY id;
+---+----+----+
| id|name| age|
+---+----+----+
|200|Mary|null|
|400| Dan|  50|
+---+----+----+

-- Function expression in `WHERE` clause.
SELECT * FROM person WHERE length(name) > 3 ORDER BY id;
+---+----+----+
| id|name| age|
+---+----+----+
|100|John|  30|
|200|Mary|null|
|300|Mike|  80|
+---+----+----+

-- `BETWEEN` expression in `WHERE` clause.
SELECT * FROM person WHERE id BETWEEN 200 AND 300 ORDER BY id;
+---+----+----+
| id|name| age|
+---+----+----+
```

```
|200|Mary|null|
|300|Mike|  80|
+---+----+----+

-- Scalar Subquery in `WHERE` clause.
SELECT * FROM person WHERE age > (SELECT avg(age) FROM person);
+---+----+---+
| id|name|age|
+---+----+---+
|300|Mike| 80|
+---+----+---+

-- Correlated Subquery in `WHERE` clause.
SELECT id FROM person
WHERE exists (SELECT id FROM person where id = 200);
+---+----+----+
|id |name|age |
+---+----+----+
|200|Mary|null|
+---+----+----+
```

Cláusula GROUP BY

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando SQL, consulte the section called “Comandos SQL compatibles”.

La cláusula `GROUP BY` se utiliza para agrupar las filas en función de un conjunto de expresiones de agrupamiento especificadas y calcular las agregaciones en el grupo de filas en función de una o más funciones de agregación especificadas.

El sistema también realiza múltiples agregaciones para el mismo conjunto de registros de entrada mediante las cláusulas `GROUPING SETS`, `CUBE`, `ROLLUP`. Las expresiones de agrupamiento y las agregaciones avanzadas se pueden mezclar en la cláusula `GROUP BY` y anidar en una cláusula `GROUPING SETS` . Consulte la sección `Mixed/Nested Grouping Analytics` para obtener más información.

Cuando se adjunta una cláusula `FILTER` a una función agregada, solo las filas coincidentes se pasan a esa función.

Sintaxis

```
GROUP BY group_expression [ , group_expression [ , ... ] ] [ WITH { ROLLUP | CUBE } ]
GROUP BY { group_expression | { ROLLUP | CUBE | GROUPING SETS } (grouping_set
 [ , ...]) } [ , ... ]
```

Mientras que las funciones agregadas se definen de la siguiente manera:

```
aggregate_name ( [ DISTINCT ] expression [ , ... ] ) [ FILTER ( WHERE
 boolean_expression ) ]
```

Parámetros

- group_expression

  Especifica los criterios en función de los cuales se agrupan las filas. La agrupación de filas se realiza en función de los valores de resultado de las expresiones de agrupación.

  Una expresión de agrupamiento puede ser un nombre de columna como `GROUP BY a`, una posición de columna como `GROUP BY 0` o una expresión similar como `GROUP BY a + b`.
- grouping_set

  Un conjunto de agrupamiento se especifica mediante cero o más expresiones separadas por comas entre paréntesis. Cuando el conjunto de agrupamiento solo tiene un elemento, se pueden omitir los paréntesis.

  Por ejemplo, `GROUPING SETS ((a), (b))` es igual que `GROUPING SETS (a, b)`.

  Sintaxis: `{ ( [ expression [ , ... ] ] ) | expression }`
- GROUPING SETS

  Agrupa las filas de cada conjunto de agrupamiento que se especifique después de `GROUPING SETS`.

  Por ejemplo, `GROUP BY GROUPING SETS ((warehouse), (product))` es semánticamente equivalente a la unión de los resultados de `GROUP BY warehouse` y `GROUP BY product`. Esta cláusula es una forma abreviada de UNION ALL, en la que cada parte del operador `UNION ALL` realiza la agregación de cada conjunto de agrupamiento especificado en la cláusula `GROUPING SETS`.

Del mismo modo, `GROUP BY GROUPING SETS ((warehouse, product), (product), ())` es semánticamente equivalente a la unión de los resultados de `GROUP BY warehouse, product`, `GROUP BY product` y un agregado global.

- ROLLUP

  Especifica varios niveles de agregación en una sentencia. Esta cláusula se utiliza para calcular las agregaciones en función de varios conjuntos de agrupamiento. `ROLLUP` es una forma abreviada de `GROUPING SETS`.

  Por ejemplo, `GROUP BY warehouse, product WITH ROLLUP` or `GROUP BY ROLLUP(warehouse, product)` equivale a `GROUP BY GROUPING SETS((warehouse, product), (warehouse), ())`.

  `GROUP BY ROLLUP(warehouse, product, (warehouse, location))` es equivalente a `GROUP BY GROUPING SETS((warehouse, product, location), (warehouse, product), (warehouse), ())`.

  Los elementos N de una especificación ROLLUP dan como resultado N+1 GROUPING SETS.

- CUBE

  La cláusula CUBE se utiliza para realizar agregaciones basadas en la combinación de columnas de agrupamiento especificadas en la cláusula GROUP BY. CUBE es una forma abreviada de GROUPING SETS.

  Por ejemplo, `GROUP BY warehouse, product WITH CUBE` or `GROUP BY CUBE(warehouse, product)` equivale a `GROUP BY GROUPING SETS((warehouse, product), (warehouse), (product), ())`.

  `GROUP BY CUBE(warehouse, product, (warehouse, location))` es equivalente a `GROUP BY GROUPING SETS((warehouse, product, location), (warehouse, product), (warehouse, location), (product, warehouse, location), (warehouse), (product), (warehouse, product), ())`. Los elementos N de una especificación `CUBE` dan como resultado 2^N `GROUPING SETS`.

- Análisis de agrupamiento mixto/anidado

  Una cláusula `GROUP BY` puede incluir múltiples group_expressions y múltiples `CUBE|ROLLUP|GROUPING SETS`. `GROUPING SETS` también puede tener cláusulas `CUBE|ROLLUP|GROUPING SETS` anidadas, como `GROUPING SETS(ROLLUP(warehouse,`

location), CUBE(warehouse, location)), GROUPING SETS(warehouse, GROUPING SETS(location, GROUPING SETS(ROLLUP(warehouse, location),, CUBE(warehouse, location)))).

CUBE|ROLLUP es solo azúcar de sintáctico para GROUPING SETS. Consulte las secciones anteriores para saber cómo traducir CUBE|ROLLUP a GROUPING SETS. group_expression se puede tratar como un solo grupo GROUPING SETS en este contexto.

En el caso de los múltiplos de GROUPING SETS en la cláusula GROUP BY, generamos un único GROUPING SETS haciendo un producto cruzado del original GROUPING SETS. Para anidar GROUPING SETS en la cláusula GROUPING SETS, simplemente cogemos sus conjuntos de agrupamiento y los eliminamos.

Por ejemplo, GROUP BY warehouse, GROUPING SETS((product), ()), GROUPING SETS((location, size), (location), (size), ()) and GROUP BY warehouse, ROLLUP(product), CUBE(location, size) equivale a GROUP BY GROUPING SETS((warehouse, product, location, size), (warehouse, product, location), (warehouse, product, size), (warehouse, product), (warehouse, location, size), (warehouse, location), (warehouse, size), (warehouse)).

GROUP BY GROUPING SETS(GROUPING SETS(warehouse), GROUPING SETS((warehouse, product))) es equivalente a GROUP BY GROUPING SETS((warehouse), (warehouse, product)).

- aggregate_name

  Especifica el nombre de una función agregada (MIN, MAX, COUNT, SUM, AVG, etc.).

- DISTINCT

  Elimina los duplicados de las filas de entrada antes de pasarlos a las funciones de agregado.

- FILTER

  Filtra las filas de entrada para las que el valor boolean_expression de la cláusula WHERE se evalúe como verdadero se pasan a la función de agregado; las demás filas se descartan.

Ejemplos

```
CREATE TABLE dealer (id INT, city STRING, car_model STRING, quantity INT);
INSERT INTO dealer VALUES
```

```
(100, 'Fremont', 'Honda Civic', 10),
(100, 'Fremont', 'Honda Accord', 15),
(100, 'Fremont', 'Honda CRV', 7),
(200, 'Dublin', 'Honda Civic', 20),
(200, 'Dublin', 'Honda Accord', 10),
(200, 'Dublin', 'Honda CRV', 3),
(300, 'San Jose', 'Honda Civic', 5),
(300, 'San Jose', 'Honda Accord', 8);

-- Sum of quantity per dealership. Group by `id`.
SELECT id, sum(quantity) FROM dealer GROUP BY id ORDER BY id;
+---+-------------+
| id|sum(quantity)|
+---+-------------+
|100|           32|
|200|           33|
|300|           13|
+---+-------------+

-- Use column position in GROUP by clause.
SELECT id, sum(quantity) FROM dealer GROUP BY 1 ORDER BY 1;
+---+-------------+
| id|sum(quantity)|
+---+-------------+
|100|           32|
|200|           33|
|300|           13|
+---+-------------+

-- Multiple aggregations.
-- 1. Sum of quantity per dealership.
-- 2. Max quantity per dealership.
SELECT id, sum(quantity) AS sum, max(quantity) AS max FROM dealer GROUP BY id ORDER BY
 id;
+---+---+---+
| id|sum|max|
+---+---+---+
|100| 32| 15|
|200| 33| 20|
|300| 13|  8|
+---+---+---+

-- Count the number of distinct dealer cities per car_model.
SELECT car_model, count(DISTINCT city) AS count FROM dealer GROUP BY car_model;
```

```
+------------+-----+
|   car_model|count|
+------------+-----+
| Honda Civic|    3|
|   Honda CRV|    2|
|Honda Accord|    3|
+------------+-----+

-- Sum of only 'Honda Civic' and 'Honda CRV' quantities per dealership.
SELECT id, sum(quantity) FILTER (
WHERE car_model IN ('Honda Civic', 'Honda CRV')
) AS `sum(quantity)` FROM dealer
GROUP BY id ORDER BY id;
+---+-------------+
| id|sum(quantity)|
+---+-------------+
|100|           17|
|200|           23|
|300|            5|
+---+-------------+

-- Aggregations using multiple sets of grouping columns in a single statement.
-- Following performs aggregations based on four sets of grouping columns.
-- 1. city, car_model
-- 2. city
-- 3. car_model
-- 4. Empty grouping set. Returns quantities for all city and car models.
SELECT city, car_model, sum(quantity) AS sum FROM dealer
GROUP BY GROUPING SETS ((city, car_model), (city), (car_model), ())
ORDER BY city;
+---------+------------+---+
|     city|   car_model|sum|
+---------+------------+---+
|     null|        null| 78|
|     null| HondaAccord| 33|
|     null|    HondaCRV| 10|
|     null|  HondaCivic| 35|
|   Dublin|        null| 33|
|   Dublin| HondaAccord| 10|
|   Dublin|    HondaCRV|  3|
|   Dublin|  HondaCivic| 20|
|  Fremont|        null| 32|
|  Fremont| HondaAccord| 15|
|  Fremont|    HondaCRV|  7|
```

```
|  Fremont|  HondaCivic| 10|
| San Jose|        null| 13|
| San Jose| HondaAccord|  8|
| San Jose|  HondaCivic|  5|
+---------+------------+---+

-- Group by processing with `ROLLUP` clause.
-- Equivalent GROUP BY GROUPING SETS ((city, car_model), (city), ())
SELECT city, car_model, sum(quantity) AS sum FROM dealer
GROUP BY city, car_model WITH ROLLUP
ORDER BY city, car_model;
+---------+------------+---+
|     city|   car_model|sum|
+---------+------------+---+
|     null|        null| 78|
|   Dublin|        null| 33|
|   Dublin| HondaAccord| 10|
|   Dublin|    HondaCRV|  3|
|   Dublin|  HondaCivic| 20|
|  Fremont|        null| 32|
|  Fremont| HondaAccord| 15|
|  Fremont|    HondaCRV|  7|
|  Fremont|  HondaCivic| 10|
| San Jose|        null| 13|
| San Jose| HondaAccord|  8|
| San Jose|  HondaCivic|  5|
+---------+------------+---+

-- Group by processing with `CUBE` clause.
-- Equivalent GROUP BY GROUPING SETS ((city, car_model), (city), (car_model), ())
SELECT city, car_model, sum(quantity) AS sum FROM dealer
GROUP BY city, car_model WITH CUBE
ORDER BY city, car_model;
+---------+------------+---+
|     city|   car_model|sum|
+---------+------------+---+
|     null|        null| 78|
|     null| HondaAccord| 33|
|     null|    HondaCRV| 10|
|     null|  HondaCivic| 35|
|   Dublin|        null| 33|
|   Dublin| HondaAccord| 10|
|   Dublin|    HondaCRV|  3|
|   Dublin|  HondaCivic| 20|
```

```
|  Fremont|        null| 32|
|  Fremont| HondaAccord| 15|
|  Fremont|    HondaCRV|  7|
|  Fremont|  HondaCivic| 10|
| San Jose|        null| 13|
| San Jose| HondaAccord|  8|
| San Jose|  HondaCivic|  5|
+---------+------------+---+

--Prepare data for ignore nulls example
CREATE TABLE person (id INT, name STRING, age INT);
INSERT INTO person VALUES
(100, 'Mary', NULL),
(200, 'John', 30),
(300, 'Mike', 80),
(400, 'Dan', 50);

--Select the first row in column age
SELECT FIRST(age) FROM person;
+--------------------+
| first(age, false)  |
+--------------------+
| NULL               |
+--------------------+

--Get the first row in column `age` ignore nulls,last row in column `id` and sum of
 column `id`.
SELECT FIRST(age IGNORE NULLS), LAST(id), SUM(id) FROM person;
+-------------------+------------------+----------+
| first(age, true)  | last(id, false)  | sum(id)  |
+-------------------+------------------+----------+
| 30                | 400              | 1000     |
+-------------------+------------------+----------+
```

Cláusula HAVING

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando SQL, consulte the section called “Comandos SQL compatibles”.

La cláusula HAVING se utiliza para filtrar los resultados generados por GROUP BY en función de la condición especificada. Suele utilizarse junto con una cláusula GROUP BY.

Sintaxis

```
HAVING boolean_expression
```

Parámetros

- boolean_expression

  Especifica cualquier expresión que se evalúe como un valor booleano. Se pueden combinar dos o más expresiones mediante los operadores lógicos (AND, OR).

  Nota: Las expresiones especificadas en la cláusula HAVING solo pueden hacer referencia a:

  1. Constantes
  2. Expresiones que aparecen en GROUP BY
  3. Funciones de agregación

Ejemplos

```
CREATE TABLE dealer (id INT, city STRING, car_model STRING, quantity INT);
INSERT INTO dealer VALUES
(100, 'Fremont', 'Honda Civic', 10),
(100, 'Fremont', 'Honda Accord', 15),
(100, 'Fremont', 'Honda CRV', 7),
(200, 'Dublin', 'Honda Civic', 20),
(200, 'Dublin', 'Honda Accord', 10),
(200, 'Dublin', 'Honda CRV', 3),
(300, 'San Jose', 'Honda Civic', 5),
(300, 'San Jose', 'Honda Accord', 8);

-- `HAVING` clause referring to column in `GROUP BY`.
SELECT city, sum(quantity) AS sum FROM dealer GROUP BY city HAVING city = 'Fremont';
+-------+---+
|   city|sum|
+-------+---+
|Fremont| 32|
+-------+---+

-- `HAVING` clause referring to aggregate function.
```

```
SELECT city, sum(quantity) AS sum FROM dealer GROUP BY city HAVING sum(quantity) > 15;
+-------+---+
|   city|sum|
+-------+---+
| Dublin| 33|
|Fremont| 32|
+-------+---+

-- `HAVING` clause referring to aggregate function by its alias.
SELECT city, sum(quantity) AS sum FROM dealer GROUP BY city HAVING sum > 15;
+-------+---+
|   city|sum|
+-------+---+
| Dublin| 33|
|Fremont| 32|
+-------+---+

-- `HAVING` clause referring to a different aggregate function than what is present in
-- `SELECT` list.
SELECT city, sum(quantity) AS sum FROM dealer GROUP BY city HAVING max(quantity) > 15;
+------+---+
|  city|sum|
+------+---+
|Dublin| 33|
+------+---+

-- `HAVING` clause referring to constant expression.
SELECT city, sum(quantity) AS sum FROM dealer GROUP BY city HAVING 1 > 0 ORDER BY city;
+--------+---+
|    city|sum|
+--------+---+
|  Dublin| 33|
| Fremont| 32|
|San Jose| 13|
+--------+---+

-- `HAVING` clause without a `GROUP BY` clause.
SELECT sum(quantity) AS sum FROM dealer HAVING sum(quantity) > 10;
+---+
|sum|
+---+
| 78|
+---+
```

Cláusula ORDER BY

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando SQL, consulte the section called “Comandos SQL compatibles”.

La cláusula `ORDER BY` se utiliza para devolver las filas de resultados ordenadas en el orden especificado por el usuario. A diferencia de la cláusula SORT BY, esta cláusula garantiza un orden total en la salida.

Sintaxis

```
ORDER BY { expression [ sort_direction | nulls_sort_order ] [ , ... ] }
```

Parámetros

- ORDER BY

  Especifica una lista de expresiones separadas por comas junto con parámetros `sort_direction` y `nulls_sort_order` opcionales que se utilizan para ordenar las filas.

- sort_direction

  De manera opcional, especifica si se deben ordenar en orden ascendente o descendente.

  Los valores válidos para la dirección de ordenación son `ASC` ascendentes y `DESC` descendentes.

  Si la dirección de ordenación no se especifica de forma explícita, las filas se ordenan por defecto en orden ascendente.

  Sintaxis: `[ ASC | DESC ]`

- nulls_sort_order

  De manera opcional, especifica si los valores `NULL` se devuelven antes o después de valores no NULL.

  Si no se especifica null_sort_order, `NULLs` ordena primero si el orden es `ASC` y NULLS ordenará en último lugar si el orden de clasificación es `DESC`.

1. Si `NULLS FIRST` se especifica, los valores NULL se devuelven primero, independientemente del orden de clasificación.

2. Si `NULLS LAST` se especifica, los valores NULL se devuelven en último lugar, independientemente del orden de clasificación.

Sintaxis: `[ NULLS { FIRST | LAST } ]`

Ejemplos

```
CREATE TABLE person (id INT, name STRING, age INT);
INSERT INTO person VALUES
(100, 'John', 30),
(200, 'Mary', NULL),
(300, 'Mike', 80),
(400, 'Jerry', NULL),
(500, 'Dan',  50);

-- Sort rows by age. By default rows are sorted in ascending manner with NULL FIRST.
SELECT name, age FROM person ORDER BY age;
+-----+----+
| name| age|
+-----+----+
|Jerry|null|
| Mary|null|
| John|  30|
|  Dan|  50|
| Mike|  80|
+-----+----+

-- Sort rows in ascending manner keeping null values to be last.
SELECT name, age FROM person ORDER BY age NULLS LAST;
+-----+----+
| name| age|
+-----+----+
| John|  30|
|  Dan|  50|
| Mike|  80|
| Mary|null|
|Jerry|null|
+-----+----+
```

```
-- Sort rows by age in descending manner, which defaults to NULL LAST.
SELECT name, age FROM person ORDER BY age DESC;
+-----+----+
| name| age|
+-----+----+
| Mike|  80|
|  Dan|  50|
| John|  30|
|Jerry|null|
| Mary|null|
+-----+----+

-- Sort rows in ascending manner keeping null values to be first.
SELECT name, age FROM person ORDER BY age DESC NULLS FIRST;
+-----+----+
| name| age|
+-----+----+
|Jerry|null|
| Mary|null|
| Mike|  80|
|  Dan|  50|
| John|  30|
+-----+----+

-- Sort rows based on more than one column with each column having different
-- sort direction.
SELECT * FROM person ORDER BY name ASC, age DESC;
+---+-----+----+
| id| name| age|
+---+-----+----+
|500|  Dan|  50|
|400|Jerry|null|
|100| John|  30|
|200| Mary|null|
|300| Mike|  80|
+---+-----+----+
```

Cláusula JOIN

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando SQL, consulte the section called “Comandos SQL compatibles”.

Una unión SQL se utiliza para combinar filas de dos relaciones en función de los criterios de unión. En la siguiente sección se describe la sintaxis general de las uniones y los distintos tipos de uniones, junto con ejemplos.

Sintaxis

```
relation INNER JOIN relation [ join_criteria ]
```

Parámetros

- relation

  Especifica la relación que se va a unir.
- join_type

  Especifica el tipo de unión.

  Sintaxis: `INNER | CROSS | LEFT OUTER`
- join_criteria

  Especifica cómo se combinarán las filas de una relación con las filas de otra relación.

  Sintaxis: `ON boolean_expression | USING ( column_name [ , ... ] )`
- boolean_expression

  Especifica una expresión con un tipo de devolución booleano.

Tipos de combinación

- Unión interna

La combinación interna debe especificarse de forma explícita. Selecciona filas que tienen valores coincidentes en ambas relaciones.

Sintaxis: `relation INNER JOIN relation [ join_criteria ]`

- Combinación izquierda

  Una combinación por la izquierda devuelve todos los valores de la relación izquierda y los valores coincidentes de la relación derecha, o agrega NULL si no hay ninguna coincidencia. También se conoce como unión externa izquierda.

  Sintaxis: `relation LEFT OUTER JOIN relation [ join_criteria ]`

- Unión cruzada

  Una unión cruzada devuelve el producto cartesiano de dos relaciones.

  Sintaxis: `relation CROSS JOIN relation [ join_criteria ]`

Ejemplos

```
-- Use employee and department tables to demonstrate different type of joins.
SELECT * FROM employee;
+---+-----+------+
| id| name|deptno|
+---+-----+------+
|105|Chloe|     5|
|103| Paul|     3|
|101| John|     1|
|102| Lisa|     2|
|104| Evan|     4|
|106|  Amy|     6|
+---+-----+------+
SELECT * FROM department;
+------+-----------+
|deptno|   deptname|
+------+-----------+
|     3|Engineering|
|     2|      Sales|
|     1|  Marketing|
+------+-----------+
```

```
-- Use employee and department tables to demonstrate inner join.
SELECT id, name, employee.deptno, deptname
FROM employee INNER JOIN department ON employee.deptno = department.deptno;
+---+-----+------+-----------|
| id| name|deptno|   deptname|
+---+-----+------+-----------|
|103| Paul|     3|Engineering|
|101| John|     1|  Marketing|
|102| Lisa|     2|      Sales|
+---+-----+------+-----------|

-- Use employee and department tables to demonstrate left join.
SELECT id, name, employee.deptno, deptname
FROM employee LEFT JOIN department ON employee.deptno = department.deptno;
+---+-----+------+-----------|
| id| name|deptno|   deptname|
+---+-----+------+-----------|
|105|Chloe|     5|       NULL|
|103| Paul|     3|Engineering|
|101| John|     1|  Marketing|
|102| Lisa|     2|      Sales|
|104| Evan|     4|       NULL|
|106|  Amy|     6|       NULL|
+---+-----+------+-----------|

-- Use employee and department tables to demonstrate cross join.
SELECT id, name, employee.deptno, deptname FROM employee CROSS JOIN department;
+---+-----+------+-----------|
| id| name|deptno|   deptname|
+---+-----+------+-----------|
|105|Chloe|     5|Engineering|
|105|Chloe|     5|  Marketing|
|105|Chloe|     5|      Sales|
|103| Paul|     3|Engineering|
|103| Paul|     3|  Marketing|
|103| Paul|     3|      Sales|
|101| John|     1|Engineering|
|101| John|     1|  Marketing|
|101| John|     1|      Sales|
|102| Lisa|     2|Engineering|
|102| Lisa|     2|  Marketing|
|102| Lisa|     2|      Sales|
|104| Evan|     4|Engineering|
|104| Evan|     4|  Marketing|
```

```
|104| Evan|     4|      Sales|
|106|  Amy|     4|Engineering|
|106|  Amy|     4|  Marketing|
|106|  Amy|     4|      Sales|
+---+-----+------+-----------+
```

Cláusula LIMIT

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando SQL, consulte the section called “Comandos SQL compatibles”.

La cláusula `LIMIT` se utiliza para restringir el número de filas devueltas por la sentencia `SELECT`. En general, esta cláusula se usa junto con `ORDER BY` para garantizar que los resultados sean deterministas.

Sintaxis

```
LIMIT { ALL | integer_expression }
```

Parámetros

- ALL

  Si se especifica, la consulta devuelve todas las filas. En otras palabras, no se aplica ningún límite si se especifica esta opción.
- integer_expression

  Especifica una expresión plegable que devuelve un entero.

Ejemplos

```
CREATE TABLE person (name STRING, age INT);
INSERT INTO person VALUES
('Jane Doe', 25),
('Pat C', 18),
('Nikki W', 16),
('John D', 25),
```

```
('Juan L', 18),
('Jorge S', 16);

-- Select the first two rows.
SELECT name, age FROM person ORDER BY name LIMIT 2;
+-------+---+
|   name|age|
+-------+---+
|  Pat C| 18|
|Jorge S| 16|
+------+---+

-- Specifying ALL option on LIMIT returns all the rows.
SELECT name, age FROM person ORDER BY name LIMIT ALL;
+--------+---+
|    name|age|
+--------+---+
|   Pat C| 18|
| Jorge S| 16|
|  Juan L| 18|
|  John D| 25|
| Nikki W| 16|
|Jane Doe| 25|
+--------+---+

-- A function expression as an input to LIMIT.
SELECT name, age FROM person ORDER BY name LIMIT length('OPENSEARCH');
+-------+---+
|   name|age|
+-------+---+
|  Pat C| 18|
|Jorge S| 16|
| Juan L| 18|
| John D| 25|
|Nikki W| 16|
+-------+---+
```

Cláusula CASE

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando SQL, consulte the section called “Comandos SQL compatibles”.

La cláusula `CASE` utiliza una regla para devolver un resultado específico en función de la condición especificada, de forma similar a las sentencias if/else de otros lenguajes de programación.

Sintaxis

```
CASE [ expression ] { WHEN boolean_expression THEN then_expression } [ ... ]
[ ELSE else_expression ]
END
```

Parámetros

- boolean_expression

  Especifica cualquier expresión que se evalúe como un valor booleano.

  Se pueden combinar dos o más expresiones mediante los operadores lógicos (`AND`, `OR`).
- then_expression

  Especifica la expresión then en función de la condición boolean_expression.

  `then_expression` y `else_expression` todas deben ser del mismo tipo o estar sujetas a un tipo común.
- else_expression

  Especifica la expresión predeterminada.

  `then_expression` y `else_expression` todas deben ser del mismo tipo o estar sujetas a un tipo común.

Ejemplos

```
CREATE TABLE person (id INT, name STRING, age INT);
```

```
INSERT INTO person VALUES
(100, 'John', 30),
(200, 'Mary', NULL),
(300, 'Mike', 80),
(400, 'Dan', 50);
SELECT id, CASE WHEN id > 200 THEN 'bigger' ELSE 'small' END FROM person;
+------+-----------------------------------------------------+
|  id  | CASE WHEN (id > 200) THEN bigger ELSE small END  |
+------+-----------------------------------------------------+
| 100  | small                                               |
| 200  | small                                               |
| 300  | bigger                                              |
| 400  | bigger                                              |
+------+-----------------------------------------------------+
SELECT id, CASE id WHEN 100 then 'bigger' WHEN  id > 300 THEN '300' ELSE 'small' END
 FROM person;
+------
+-------------------------------------------------------------------------------------------------------
+
|  id  | CASE WHEN (id = 100) THEN bigger WHEN (id = CAST((id > 300) AS INT)) THEN 300
 ELSE small END  |
+------
+-------------------------------------------------------------------------------------------------------
+
| 100  | bigger
                |
| 200  | small
                |
| 300  | small
                |
| 400  | small
                |
+------
+-------------------------------------------------------------------------------------------------------
+
```

Expresiones de tabla comunes

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando SQL, consulte the section called “Comandos SQL compatibles”.

Una expresión de tabla común (CTE) define un conjunto de resultados temporal al que un usuario puede hacer referencia posiblemente varias veces dentro del ámbito de una sentencia SQL. Una CTE se utiliza principalmente en una declaración `SELECT`.

Sintaxis

```
WITH common_table_expression [ , ... ]
```

Mientras que `common_table_expression` se define como:

```
Syntexpression_name [ ( column_name [ , ... ] ) ] [ AS ] ( query )
```

Parámetros

- expression_name

  Especifica un nombre para la expresión de tabla común.
- query

  Una sentencia `SELECT`.

Ejemplos

```
-- CTE with multiple column aliases
WITH t(x, y) AS (SELECT 1, 2)
SELECT * FROM t WHERE x = 1 AND y = 2;
+---+---+
|  x|  y|
+---+---+
|  1|  2|
+---+---+

-- CTE in CTE definition
WITH t AS (
WITH t2 AS (SELECT 1)
SELECT * FROM t2
)
SELECT * FROM t;
+---+
|  1|
+---+
```

```
|  1|
+---+

-- CTE in subquery
SELECT max(c) FROM (
WITH t(c) AS (SELECT 1)
SELECT * FROM t
);
+------+
|max(c)|
+------+
|     1|
+------+

-- CTE in subquery expression
SELECT (
WITH t AS (SELECT 1)
SELECT * FROM t
);
+----------------+
|scalarsubquery()|
+----------------+
|               1|
+----------------+

-- CTE in CREATE VIEW statement
CREATE VIEW v AS
WITH t(a, b, c, d) AS (SELECT 1, 2, 3, 4)
SELECT * FROM t;
SELECT * FROM v;
+---+---+---+---+
|  a|  b|  c|  d|
+---+---+---+---+
|  1|  2|  3|  4|
+---+---+---+---+
```

EXPLAIN

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando SQL, consulte the section called “Comandos SQL compatibles”.

La sentencia EXPLAIN se utiliza para proporcionar planes lógicos/físicos para una sentencia de entrada. De forma predeterminada, esta cláusula proporciona información únicamente sobre un plan físico.

Sintaxis

```
EXPLAIN [ EXTENDED | CODEGEN | COST | FORMATTED ] statement
```

Parámetros

- EXTENDED

  Genera un plan lógico diseccionado, un plan lógico analizado, un plan lógico optimizado y un plan físico.

  El plan lógico diseccionado es un plan sin resolver que se extrajo de la consulta.

  Los planes lógicos analizados se transforman y traducen `unresolvedAttribute` y `unresolvedRelation` a objetos completamente tipificados.

  El plan lógico optimizado se transforma mediante un conjunto de reglas de optimización, lo que da como resultado el plan físico.
- CODEGEN

  Genera el código para la declaración, si la hubiera y un plan físico.
- COST

  Si las estadísticas de los nodos del plan están disponibles, genera un plan lógico y las estadísticas.
- FORMATTED

  Genera dos secciones: un esquema físico del plan y detalles de los nodos.
- statement

  Especifica una sentencia SQL que se va a explicar.

Ejemplos

```
-- Default Output
EXPLAIN select k, sum(v) from values (1, 2), (1, 3) t(k, v) group by k;
+-------------------------------------------------+
```

```
|                                                    plan|
+-------------------------------------------------------+
| == Physical Plan ==
*(2) HashAggregate(keys=[k#33], functions=[sum(cast(v#34 as bigint))])
+- Exchange hashpartitioning(k#33, 200), true, [id=#59]
+- *(1) HashAggregate(keys=[k#33], functions=[partial_sum(cast(v#34 as bigint))])
+- *(1) LocalTableScan [k#33, v#34]
|
+-------------------------------------------------------

-- Using Extended
EXPLAIN EXTENDED select k, sum(v) from values (1, 2), (1, 3) t(k, v) group by k;
+-------------------------------------------------------+
|                                                    plan|
+-------------------------------------------------------+
| == Parsed Logical Plan ==
'Aggregate ['k], ['k, unresolvedalias('sum('v), None)]
 +- 'SubqueryAlias `t`
+- 'UnresolvedInlineTable [k, v], [List(1, 2), List(1, 3)]

 == Analyzed Logical Plan ==
 k: int, sum(v): bigint
 Aggregate [k#47], [k#47, sum(cast(v#48 as bigint)) AS sum(v)#50L]
 +- SubqueryAlias `t`
    +- LocalRelation [k#47, v#48]

 == Optimized Logical Plan ==
 Aggregate [k#47], [k#47, sum(cast(v#48 as bigint)) AS sum(v)#50L]
 +- LocalRelation [k#47, v#48]

 == Physical Plan ==
 *(2) HashAggregate(keys=[k#47], functions=[sum(cast(v#48 as bigint))], output=[k#47,
 sum(v)#50L])
+- Exchange hashpartitioning(k#47, 200), true, [id=#79]
   +- *(1) HashAggregate(keys=[k#47], functions=[partial_sum(cast(v#48 as bigint))],
 output=[k#47, sum#52L])
    +- *(1) LocalTableScan [k#47, v#48]
|
+-------------------------------------------------------+

-- Using Formatted
EXPLAIN FORMATTED select k, sum(v) from values (1, 2), (1, 3) t(k, v) group by k;
+-------------------------------------------------------+
|                                                    plan|
```

```
+-----------------------------------------------------+
| == Physical Plan ==
 * HashAggregate (4)
 +- Exchange (3)
    +- * HashAggregate (2)
       +- * LocalTableScan (1)


 (1) LocalTableScan [codegen id : 1]
 Output: [k#19, v#20]

 (2) HashAggregate [codegen id : 1]
 Input: [k#19, v#20]

 (3) Exchange
 Input: [k#19, sum#24L]

 (4) HashAggregate [codegen id : 2]
 Input: [k#19, sum#24L]
|
+-----------------------------------------------------+
```

Cláusula LATERAL SUBQUERY

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando SQL, consulte the section called “Comandos SQL compatibles”.

`LATERAL SUBQUERY` es una subconsulta precedida de la palabra clave `LATERAL`. Proporciona una forma de hacer referencia a las columnas de la cláusula `FROM` anterior. Sin la palabra clave `LATERAL`, las subconsultas solo pueden hacer referencia a las columnas de la consulta externa, pero no a las de la cláusula `FROM`. `LATERAL SUBQUERY` simplifica y hace que las consultas complicadas sean más sencillas y eficaces.

Sintaxis

```
[ LATERAL ] primary_relation [ join_relation ]
```

Parámetros

- primary_relation

  Especifica la relación principal. Puede ser una de las siguientes opciones:

  1. Relación de tabla
  2. Consulta con alias

     Sintaxis: `( query ) [ [ AS ] alias ]`
  3. Relación con alias

     `Syntax: ( relation ) [ [ AS ] alias ]`

Ejemplos

```
CREATE TABLE t1 (c1 INT, c2 INT);
INSERT INTO t1 VALUES (0, 1), (1, 2);
CREATE TABLE t2 (c1 INT, c2 INT);
INSERT INTO t2 VALUES (0, 2), (0, 3);
SELECT * FROM t1,
LATERAL (SELECT * FROM t2 WHERE t1.c1 = t2.c1);
+--------+-------+--------+-------+
|  t1.c1 | t1.c2 |  t2.c1 | t2.c2 |
+-------+--------+--------+-------+
|    0   |   1   |    0   |   3   |
|    0   |   1   |    0   |   2   |
+-------+--------+--------+-------+
SELECT a, b, c FROM t1,
LATERAL (SELECT c1 + c2 AS a),
LATERAL (SELECT c1 - c2 AS b),
LATERAL (SELECT a * b AS c);
+--------+-------+--------+
|    a   |   b   |    c   |
+-------+--------+--------+
|    3   |  -1   |   -3   |
|    1   |  -1   |   -1   |
+-------+--------+--------+
```

Cláusula LATERAL VIEW

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando SQL, consulte the section called “Comandos SQL compatibles”.

La cláusula `LATERAL VIEW` se usa junto con funciones generadoras, por ejemplo `EXPLODE`, que generarán una tabla virtual que contiene una o más filas. `LATERAL VIEW` aplicará las filas a cada fila de salida original.

Sintaxis

```
LATERAL VIEW [ OUTER ] generator_function ( expression [ , ... ] ) [ table_alias ] AS
 column_alias [ , ... ]
```

Parámetros

- OUTER

  Si `OUTER` se especifica, devuelve null si una matriz/mapa de entrada está vacía o es nula.
- generator_function

  Especifica una función generadora (`EXPLODE`, `INLINE`, etc.).
- table_alias

  El alias de `generator_function`, que es opcional.
- column_alias

  Muestra los alias de las columnas de `generator_function`, que se pueden usar en las filas de salida.

  Puede tener varios alias si `generator_function` tiene varias columnas de salida.

Ejemplos

```
CREATE TABLE person (id INT, name STRING, age INT, class INT, address STRING);
INSERT INTO person VALUES
(100, 'John', 30, 1, 'Street 1'),
```

```
(200, 'Mary', NULL, 1, 'Street 2'),
(300, 'Mike', 80, 3, 'Street 3'),
(400, 'Dan', 50, 4, 'Street 4');
SELECT * FROM person
LATERAL VIEW EXPLODE(ARRAY(30, 60)) tableName AS c_age
LATERAL VIEW EXPLODE(ARRAY(40, 80)) AS d_age;
+------+-------+-------+--------+-----------+--------+--------+
|  id  | name  |  age  | class  |  address  | c_age  | d_age  |
+------+-------+-------+--------+-----------+--------+--------+
| 100  | John  | 30    | 1      | Street 1  | 30     | 40     |
| 100  | John  | 30    | 1      | Street 1  | 30     | 80     |
| 100  | John  | 30    | 1      | Street 1  | 60     | 40     |
| 100  | John  | 30    | 1      | Street 1  | 60     | 80     |
| 200  | Mary  | NULL  | 1      | Street 2  | 30     | 40     |
| 200  | Mary  | NULL  | 1      | Street 2  | 30     | 80     |
| 200  | Mary  | NULL  | 1      | Street 2  | 60     | 40     |
| 200  | Mary  | NULL  | 1      | Street 2  | 60     | 80     |
| 300  | Mike  | 80    | 3      | Street 3  | 30     | 40     |
| 300  | Mike  | 80    | 3      | Street 3  | 30     | 80     |
| 300  | Mike  | 80    | 3      | Street 3  | 60     | 40     |
| 300  | Mike  | 80    | 3      | Street 3  | 60     | 80     |
| 400  | Dan   | 50    | 4      | Street 4  | 30     | 40     |
| 400  | Dan   | 50    | 4      | Street 4  | 30     | 80     |
| 400  | Dan   | 50    | 4      | Street 4  | 60     | 40     |
| 400  | Dan   | 50    | 4      | Street 4  | 60     | 80     |
+------+-------+-------+--------+-----------+--------+--------+
SELECT c_age, COUNT(1) FROM person
LATERAL VIEW EXPLODE(ARRAY(30, 60)) AS c_age
LATERAL VIEW EXPLODE(ARRAY(40, 80)) AS d_age
GROUP BY c_age;
+--------+-----------+
| c_age  | count(1)  |
+--------+-----------+
| 60     | 8         |
| 30     | 8         |
+--------+-----------+
SELECT * FROM person
LATERAL VIEW EXPLODE(ARRAY()) tableName AS c_age;
+-----+-------+------+--------+----------+--------+
| id  | name  | age  | class  | address  | c_age  |
+-----+-------+------+--------+----------+--------+
+-----+-------+------+--------+----------+--------+
SELECT * FROM person
LATERAL VIEW OUTER EXPLODE(ARRAY()) tableName AS c_age;
```

```
+------+-------+-------+--------+-----------+--------+
|  id  | name  |  age  | class  |  address  | c_age  |
+------+-------+-------+--------+-----------+--------+
| 100  | John  | 30    | 1      | Street 1  | NULL   |
| 200  | Mary  | NULL  | 1      | Street 2  | NULL   |
| 300  | Mike  | 80    | 3      | Street 3  | NULL   |
| 400  | Dan   | 50    | 4      | Street 4  | NULL   |
+------+-------+-------+--------+-----------+--------+
```

Predicado LIKE

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando SQL, consulte the section called “Comandos SQL compatibles”.

Se utiliza un predicado `LIKE` para buscar un patrón específico. Este predicado también admite varios patrones con cuantificadores como `ANY`, `SOME` y `ALL`.

Sintaxis

```
[ NOT ] { LIKE search_pattern [ ESCAPE esc_char ] | [ RLIKE | REGEXP ] regex_pattern }
[ NOT ] { LIKE quantifiers ( search_pattern [ , ... ]) }
```

Parámetros

- search_pattern

  Especifica un patrón de cadenas que se va a buscar mediante la cláusula LIKE. Puede contener caracteres especiales que coincidan con el patrón:

  - `%` coincide con 0 o más caracteres.
  - `_` coincide exactamente con un carácter.
- esc_char

  Especifica el carácter de escape. El carácter de escape predeterminado es `\`.
- regex_pattern

  Especifica un patrón de búsqueda de expresiones regulares que se va a buscar mediante la cláusula `RLIKE` o `REGEXP`.

- quantifiers

  Especifica los cuantificadores de predicados que incluyen ANY, SOME y ALL.

  ANY o SOME significan que si uno de los patrones coincide con la entrada, devuelve verdadero.

  ALL significa que si todos los patrones coinciden con la entrada, entonces devuelve verdadero.

Ejemplos

```
CREATE TABLE person (id INT, name STRING, age INT);
INSERT INTO person VALUES
(100, 'John', 30),
(200, 'Mary', NULL),
(300, 'Mike', 80),
(400, 'Dan',  50),
(500, 'Evan_w', 16);
SELECT * FROM person WHERE name LIKE 'M%';
+---+----+----+
| id|name| age|
+---+----+----+
|300|Mike|  80|
|200|Mary|null|
+---+----+----+
SELECT * FROM person WHERE name LIKE 'M_ry';
+---+----+----+
| id|name| age|
+---+----+----+
|200|Mary|null|
+---+----+----+
SELECT * FROM person WHERE name NOT LIKE 'M_ry';
+---+------+---+
| id|  name|age|
+---+------+---+
|500|Evan_W| 16|
|300|  Mike| 80|
|100|  John| 30|
|400|   Dan| 50|
+---+------+---+
SELECT * FROM person WHERE name RLIKE 'M+';
+---+----+----+
| id|name| age|
+---+----+----+
```

```
|300|Mike|  80|
|200|Mary|null|
+---+----+----+
SELECT * FROM person WHERE name REGEXP 'M+';
+---+----+----+
| id|name| age|
+---+----+----+
|300|Mike|  80|
|200|Mary|null|
+---+----+----+
SELECT * FROM person WHERE name LIKE '%\_%';
+---+------+---+
| id|  name|age|
+---+------+---+
|500|Evan_W| 16|
+---+------+---+
SELECT * FROM person WHERE name LIKE '%$_%' ESCAPE '$';
+---+------+---+
| id|  name|age|
+---+------+---+
|500|Evan_W| 16|
+---+------+---+
SELECT * FROM person WHERE name LIKE ALL ('%an%', '%an');
+---+----+----+
| id|name| age|
+---+----+----+
|400| Dan|  50|
+---+----+----+
SELECT * FROM person WHERE name LIKE ANY ('%an%', '%an');
+---+------+---+
| id|  name|age|
+---+------+---+
|400|   Dan| 50|
|500|Evan_W| 16|
+---+------+---+
SELECT * FROM person WHERE name LIKE SOME ('%an%', '%an');
+---+------+---+
| id|  name|age|
+---+------+---+
|400|   Dan| 50|
|500|Evan_W| 16|
+---+------+---+
SELECT * FROM person WHERE name NOT LIKE ALL ('%an%', '%an');
+---+----+----+
```

```
| id|name| age|
+---+----+----+
|100|John|  30|
|200|Mary|null|
|300|Mike|  80|
+---+----+----+
SELECT * FROM person WHERE name NOT LIKE ANY ('%an%', '%an');
+---+------+----+
| id|  name| age|
+---+------+----+
|100|  John|  30|
|200|  Mary|null|
|300|  Mike|  80|
|500|Evan_W|  16|
+---+------+----+
SELECT * FROM person WHERE name NOT LIKE SOME ('%an%', '%an');
+---+------+----+
| id|  name| age|
+---+------+----+
|100|  John|  30|
|200|  Mary|null|
|300|  Mike|  80|
|500|Evan_W|  16|
+---+------+----+
```

OFFSET

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando SQL, consulte the section called “Comandos SQL compatibles”.

La cláusula `OFFSET` se utiliza para especificar el número de filas que se van a omitir antes de empezar a devolver las filas devueltas por la sentencia `SELECT`. En general, esta cláusula se utiliza en conjunto con `ORDER BY` para garantizar que los resultados sean deterministas.

Sintaxis

```
OFFSET integer_expression
```

Parámetros

- integer_expression

  Especifica una expresión plegable que devuelve un entero.

Ejemplos

```
CREATE TABLE person (name STRING, age INT);
INSERT INTO person VALUES
('Jane Doe', 25),
('Pat C', 18),
('Nikki W', 16),
('Juan L', 25),
('John D', 18),
('Jorge S', 16);

-- Skip the first two rows.
SELECT name, age FROM person ORDER BY name OFFSET 2;
+-------+---+
|   name|age|
+-------+---+
| John D| 18|
| Juan L| 25|
|Nikki W| 16|
|Jane Doe| 25|
+-------+---+

-- Skip the first two rows and returns the next three rows.
SELECT name, age FROM person ORDER BY name LIMIT 3 OFFSET 2;
+-------+---+
|   name|age|
+-------+---+
| John D| 18|
| Juan L| 25|
|Nikki W| 16|
+-------+---+

-- A function expression as an input to OFFSET.
SELECT name, age FROM person ORDER BY name OFFSET length('WAGON');
+-------+---+
|   name|age|
+-------+---+
|Jane Doe| 25|
```

```
+-------+---+
```

Cláusula PIVOT

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando SQL, consulte the section called “Comandos SQL compatibles”.

La cláusula `PIVOT` se utiliza desde la perspectiva de los datos. Podemos obtener los valores agregados en función de valores de columna específicos, que se convertirán en varias columnas utilizadas en la cláusula `SELECT`. La cláusula `PIVOT` se puede especificar después del nombre de la tabla o la subconsulta.

Sintaxis

```
PIVOT ( { aggregate_expression [ AS aggregate_expression_alias ] } [ , ... ] FOR
 column_list IN ( expression_list ) )
```

Parámetros

- expresión_de_agregación

  Especifica una expresión agregada (`SUM(a)`, `COUNT(DISTINCT b)`, etc.

- aggregate_expression_alias

  Especifica un alias para la expresión agregada.

- column_list

  Contiene columnas en la cláusula `FROM`, que especifica las columnas que desea reemplazar por columnas nuevas. Puede utilizar corchetes para rodear las columnas, por ejemplo `(c1, c2)`.

- expression_list

  Especifica las columnas nuevas, que se utilizan para hacer coincidir los valores en `column_list` como la condición de agregación. También puede usar alias para los bots.

Ejemplos

```
CREATE TABLE person (id INT, name STRING, age INT, class INT, address STRING);
INSERT INTO person VALUES
(100, 'John', 30, 1, 'Street 1'),
(200, 'Mary', NULL, 1, 'Street 2'),
(300, 'Mike', 80, 3, 'Street 3'),
(400, 'Dan', 50, 4, 'Street 4');
SELECT * FROM person
PIVOT (
SUM(age) AS a, AVG(class) AS c
FOR name IN ('John' AS john, 'Mike' AS mike)
);
+------+-----------+---------+---------+---------+---------+
|  id  |  address  | john_a  | john_c  | mike_a  | mike_c  |
+------+-----------+---------+---------+---------+---------+
| 200  | Street 2  | NULL    | NULL    | NULL    | NULL    |
| 100  | Street 1  | 30      | 1.0     | NULL    | NULL    |
| 300  | Street 3  | NULL    | NULL    | 80      | 3.0     |
| 400  | Street 4  | NULL    | NULL    | NULL    | NULL    |
+------+-----------+---------+---------+---------+---------+
SELECT * FROM person
PIVOT (
SUM(age) AS a, AVG(class) AS c
FOR (name, age) IN (('John', 30) AS c1, ('Mike', 40) AS c2)
);
+------+-----------+-------+-------+-------+-------+
|  id  |  address  | c1_a  | c1_c  | c2_a  | c2_c  |
+------+-----------+-------+-------+-------+-------+
| 200  | Street 2  | NULL  | NULL  | NULL  | NULL  |
| 100  | Street 1  | 30    | 1.0   | NULL  | NULL  |
| 300  | Street 3  | NULL  | NULL  | NULL  | NULL  |
| 400  | Street 4  | NULL  | NULL  | NULL  | NULL  |
+------+-----------+-------+-------+-------+-------+
```

Operadores de establecimiento

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando SQL, consulte the section called “Comandos SQL compatibles”.

Los operadores de conjuntos se utilizan para combinar dos relaciones de entrada en una sola. OpenSearch SQL admite tres tipos de operadores de conjuntos:

- `EXCEPT` o `MINUS`
- `INTERSECT`
- `UNION`

Las relaciones de entrada deben tener el mismo número de columnas y tipos de datos compatibles para las columnas respectivas.

EXCEPT

`EXCEPT` y `EXCEPT ALL` devuelven las filas que se encuentran en una relación pero no en la otra. `EXCEPT` (alternativamente, `EXCEPT DISTINCT`) solo toma filas distintas, pero `EXCEPT ALL` no elimina los duplicados de las filas de resultados. Tenga en cuenta que `MINUS` es un alias de `EXCEPT`.

Sintaxis

```
[ ( ] relation [ ) ] EXCEPT | MINUS [ ALL | DISTINCT ] [ ( ] relation [ ) ]
```

Ejemplos

```
-- Use table1 and table2 tables to demonstrate set operators in this page.
SELECT * FROM table1;
+---+
|  c|
+---+
|  3|
|  1|
|  2|
|  2|
|  3|
|  4|
+---+
SELECT * FROM table2;
+---+
|  c|
+---+
|  5|
|  1|
```

```
|  2|
|  2|
+---+
SELECT c FROM table1 EXCEPT SELECT c FROM table2;
+---+
|  c|
+---+
|  3|
|  4|
+---+
SELECT c FROM table1 MINUS SELECT c FROM table2;
+---+
|  c|
+---+
|  3|
|  4|
+---+
SELECT c FROM table1 EXCEPT ALL (SELECT c FROM table2);
+---+
|  c|
+---+
|  3|
|  3|
|  4|
+---+
SELECT c FROM table1 MINUS ALL (SELECT c FROM table2);
+---+
|  c|
+---+
|  3|
|  3|
|  4|
+---+
```

INTERSECT

`INTERSECT` y `INTERSECT ALL` devuelven las filas que se encuentran en ambas relaciones. `INTERSECT` (alternativamente, `INTERSECT DISTINCT`) solo toma filas distintas, pero `INTERSECT ALL` no elimina los duplicados de las filas de resultados.

Sintaxis

```
[ ( ] relation [ ) ] INTERSECT [ ALL | DISTINCT ] [ ( ] relation [ ) ]
```

Ejemplos

```
(SELECT c FROM table1) INTERSECT (SELECT c FROM table2);
+---+
|  c|
+---+
|  1|
|  2|
+---+
(SELECT c FROM table1) INTERSECT DISTINCT (SELECT c FROM table2);
+---+
|  c|
+---+
|  1|
|  2|
+---+
(SELECT c FROM table1) INTERSECT ALL (SELECT c FROM table2);
+---+
|  c|
+---+
|  1|
|  2|
|  2|
+---+
```

UNION

`UNION` y `UNION ALL` devuelven las filas que se encuentran en cualquiera de las dos relaciones. `UNION` (alternativamente, `UNION DISTINCT`) solo toma filas distintas, pero `UNION ALL` no elimina los duplicados de las filas de resultados.

Sintaxis

```
 [ ( ] relation [ ) ] UNION [ ALL | DISTINCT ] [ ( ] relation [ ) ]
```

Ejemplos

```
(SELECT c FROM table1) UNION (SELECT c FROM table2);
+---+
|  c|
+---+
|  1|
|  3|
```

```
|  5|
|  4|
|  2|
+---+
(SELECT c FROM table1) UNION DISTINCT (SELECT c FROM table2);
+---+
|  c|
+---+
|  1|
|  3|
|  5|
|  4|
|  2|
+---+
SELECT c FROM table1 UNION ALL (SELECT c FROM table2);
+---+
|  c|
+---+
|  3|
|  1|
|  2|
|  2|
|  3|
|  4|
|  5|
|  1|
|  2|
|  2|
+---+
```

Cláusula SORT BY

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando SQL, consulte the section called “Comandos SQL compatibles”.

La cláusula `SORT BY` se utiliza para devolver las filas de resultados ordenadas dentro de cada partición en el orden especificado por el usuario. Cuando hay más de una partición, `SORT BY` puede devolver un resultado parcialmente ordenado. Esto es diferente a la cláusula `ORDER BY` que garantiza un orden total de la salida.

Sintaxis

```
SORT BY { expression [ sort_direction | nulls_sort_order ] [ , ... ] }
```

Parámetros

- SORT BY

  Especifica una lista de expresiones separadas por comas junto con los parámetros opcionales sort_direction y nulls_sort_order que se utilizan para ordenar las filas de cada partición.
- sort_direction

  De manera opcional, especifica si se deben ordenar en orden ascendente o descendente.

  Los valores válidos para la dirección de ordenación son `ASC` ascendentes y `DESC` descendentes.

  Si la dirección de ordenación no se especifica de forma explícita, las filas se ordenan por defecto en orden ascendente.

  Sintaxis: `[ ASC | DESC ]`
- nulls_sort_order

  De manera opcional, especifica si los valores NULL se devuelven antes o después de los valores no NULL.

  Si `null_sort_order` no se especifica, los valores NULL ordenan primero si el orden de clasificación es `ASC` y los NULL ordenan al final si el orden es `DESC`.

  1. Si `NULLS FIRST` se especifica, los valores NULL se devuelven primero, independientemente del orden de clasificación.

  2. Si `NULLS LAST` se especifica, los valores NULL se devuelven en último lugar, independientemente del orden de clasificación.

  Sintaxis: `[ NULLS { FIRST | LAST } ]`

Ejemplos

```
CREATE TABLE person (zip_code INT, name STRING, age INT);
INSERT INTO person VALUES
```

```
(94588, 'Shirley Rodriguez', 50),
(94588, 'Juan Li', 18),
(94588, 'Anil K', 27),
(94588, 'John D', NULL),
(94511, 'David K', 42),
(94511, 'Aryan B.', 18),
(94511, 'Lalit B.', NULL);
-- Sort rows by `name` within each partition in ascending manner
SELECT name, age, zip_code FROM person SORT BY name;
+------------------+----+--------+
|              name| age|zip_code|
+------------------+----+--------+
|            Anil K|  27|   94588|
|           Juan Li|  18|   94588|
|            John D|null|   94588|
| Shirley Rodriguez|  50|   94588|
|          Aryan B.|  18|   94511|
|           David K|  42|   94511|
|          Lalit B.|null|   94511|
+------------------+----+--------+
-- Sort rows within each partition using column position.
SELECT name, age, zip_code FROM person SORT BY 1;
+----------------+----+----------+
|              name| age|zip_code|
+------------------+----+--------+
|            Anil K|  27|   94588|
|           Juan Li|  18|   94588|
|            John D|null|   94588|
| Shirley Rodriguez|  50|   94588|
|          Aryan B.|  18|   94511|
|           David K|  42|   94511|
|          Lalit B.|null|   94511|
+------------------+----+--------+

-- Sort rows within partition in ascending manner keeping null values to be last.
SELECT age, name, zip_code FROM person SORT BY age NULLS LAST;
+----+-----------------+--------+
| age|             name|zip_code|
+----+-----------------+--------+
|  18|          Juan Li|   94588|
|  27|           Anil K|   94588|
|  50|Shirley Rodriguez|   94588|
|null|           John D|   94588|
|  18|         Aryan B.|   94511|
```

```
|  42|           David K|   94511|
|null|          Lalit B.|   94511|
+----------+--------+--------+

-- Sort rows by age within each partition in descending manner, which defaults to NULL
 LAST.
SELECT age, name, zip_code FROM person SORT BY age DESC;
+----+------------------+--------+
| age|              name|zip_code|
+----+------------------+--------+
|  50|          Shirley Rodriguez|   94588|
|  27|            Anil K|   94588|
|  18|           Juan Li|   94588|
|null|            John D|   94588|
|  42|           David K|   94511|
|  18|          Aryan B.|   94511|
|null|          Lalit B.|   94511|
+----+------------------+--------+

-- Sort rows by age within each partition in descending manner keeping null values to
 be first.
SELECT age, name, zip_code FROM person SORT BY age DESC NULLS FIRST;
+----+------------------+--------+
| age|              name|zip_code|
+----+------------------+--------+
|null|            John D|   94588|
|  50| Shirley Rodriguez|   94588|
|  27|            Anil K|   94588|
|  18|           Juan Li|   94588|
|null|          Lalit B.|   94511|
|  42|           David K|   94511|
|  18|          Aryan B.|   94511|
+----+------------------+--------+

-- Sort rows within each partition based on more than one column with each column
 having
-- different sort direction.
SELECT name, age, zip_code FROM person
SORT BY name ASC, age DESC;
+------------------+----+--------+
|              name| age|zip_code|
+------------------+----+--------+
|            Anil K|  27|   94588|
|           Juan Li|  18|   94588|
```

```
|            John D|null|   94588|
| Shirley Rodriguez|  50|   94588|
|          Aryan B.|  18|   94511|
|           David K|  42|   94511|
|          Lalit B.|null|   94511|
+------------------+----+--------+
```

UNPIVOT

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando SQL, consulte the section called “Comandos SQL compatibles”.

La cláusula `UNPIVOT` transforma varias columnas en varias filas utilizadas en la cláusula `SELECT`. La cláusula `UNPIVOT` se puede especificar después del nombre de la tabla o la subconsulta.

Sintaxis

```
UNPIVOT [ { INCLUDE | EXCLUDE } NULLS ] (
    { single_value_column_unpivot | multi_value_column_unpivot }
) [[AS] alias]

single_value_column_unpivot:
    values_column
    FOR name_column
    IN (unpivot_column [[AS] alias] [, ...])

multi_value_column_unpivot:
    (values_column [, ...])
    FOR name_column
    IN ((unpivot_column [, ...]) [[AS] alias] [, ...])
```

Parámetros

- unpivot_column

  Contiene las columnas de la `FROM` cláusula, que especifican las columnas que queremos dejar de pivotar.
- name_column

El nombre de la columna que contiene los nombres de las columnas no pivotantes.

- values_column

  El nombre de la columna que contiene los valores de las columnas no pivotantes.

Ejemplos

```
CREATE TABLE sales_quarterly (year INT, q1 INT, q2 INT, q3 INT, q4 INT);
INSERT INTO sales_quarterly VALUES
(2020, null, 1000, 2000, 2500),
(2021, 2250, 3200, 4200, 5900),
(2022, 4200, 3100, null, null);
-- column names are used as unpivot columns
SELECT * FROM sales_quarterly
UNPIVOT (
sales FOR quarter IN (q1, q2, q3, q4)
);
+------+---------+-------+
| year | quarter | sales |
+------+---------+-------+
| 2020 | q2      | 1000  |
| 2020 | q3      | 2000  |
| 2020 | q4      | 2500  |
| 2021 | q1      | 2250  |
| 2021 | q2      | 3200  |
| 2021 | q3      | 4200  |
| 2021 | q4      | 5900  |
| 2022 | q1      | 4200  |
| 2022 | q2      | 3100  |
+------+---------+-------+
-- NULL values are excluded by default, they can be included
-- unpivot columns can be alias
-- unpivot result can be referenced via its alias
SELECT up.* FROM sales_quarterly
UNPIVOT INCLUDE NULLS (
sales FOR quarter IN (q1 AS Q1, q2 AS Q2, q3 AS Q3, q4 AS Q4)
) AS up;
+------+---------+-------+
| year | quarter | sales |
+------+---------+-------+
| 2020 | Q1      | NULL  |
| 2020 | Q2      | 1000  |
```

```
| 2020 | Q3      | 2000  |
| 2020 | Q4      | 2500  |
| 2021 | Q1      | 2250  |
| 2021 | Q2      | 3200  |
| 2021 | Q3      | 4200  |
| 2021 | Q4      | 5900  |
| 2022 | Q1      | 4200  |
| 2022 | Q2      | 3100  |
| 2022 | Q3      | NULL  |
| 2022 | Q4      | NULL  |
+------+---------+-------+
-- multiple value columns can be unpivoted per row
SELECT * FROM sales_quarterly
UNPIVOT EXCLUDE NULLS (
(first_quarter, second_quarter)
FOR half_of_the_year IN (
(q1, q2) AS H1,
(q3, q4) AS H2
)
);
+------+------------------+---------------+----------------+
|  id  | half_of_the_year | first_quarter | second_quarter |
+------+------------------+---------------+----------------+
| 2020 | H1               | NULL          | 1000           |
| 2020 | H2               | 2000          | 2500           |
| 2021 | H1               | 2250          | 3200           |
| 2021 | H2               | 4200          | 5900           |
| 2022 | H1               | 4200          | 3100           |
+------+------------------+---------------+----------------+
```

Comandos compatibles PPL

En las siguientes tablas, se muestran qué comandos PPL son compatibles con OpenSearch Dashboards para consultar Registros de CloudWatch, Amazon S3 o Security Lake y qué comandos son compatibles con Información de registros de CloudWatch. Información de registros de CloudWatch utiliza la misma sintaxis de PPL que OpenSearch Dashboards al consultar Registros de CloudWatch y en las tablas se hace referencia a ambos como Registros de CloudWatch.

> **Note**
>
> Al analizar datos fuera de OpenSearch Service, es posible que los comandos se ejecuten de forma diferente a como lo hacen en los índices de OpenSearch.

Temas

- Comandos
- Funciones
- Información adicional para los usuarios de Información de registros de CloudWatch que utilizan OpenSearch PPL

Comandos

Comando PPL	Descripción	Registros de CloudWatch	Amazon S3	Security Lake	Comando de ejemplo
the section called “campos”	Muestra un conjunto de campos que necesitan ser proyectados.	compatible	compatible	compatible	`fields field1, field2`
the section called “where”	Filtra los datos en función de las condiciones que especifiq ue.	compatible	compatible	compatible	`where field1="success" | where field2 != "i-023fe0a90929d8822" | fields field3, col4, col5, col6 | head 1000`

Comando PPL	Descripción	Registros de CloudWatch	Amazon S3	Security Lake	Comando de ejemplo
the section called “stats”	Realiza agregaciones y cálculos.	compatible	compatible	compatible	`stats count(), count(`field1`), min(`field1`), max(`field1`), avg(`field1`) by field2 | head 1000`
the section called “parse”	Extrae un patrón de expresión regular (regex) de una cadena y muestra el patrón extraído. El patrón extraído se puede utilizar además para crear nuevos campos o filtrar datos.	compatible	compatible	compatible	`parse `field1` ".*/(?<field2>[^/]+$)" | where field2 = "requestId" | fields field2, `field2` | head 1000`

Comando PPL	Descripción	Registros de CloudWatch	Amazon S3	Security Lake	Comando de ejemplo
the section called “patrones”	Extrae los patrones de registro de un campo de texto y agrega los resultados al resultado de la búsqueda. Al agrupar los registros por sus patrones, resulta más fácil agregar estadísticas de grandes volúmenes de datos de registro para analizarlos y solucionar problemas.	no compatible	compatible	compatible	`patterns new_field='no_numbers' pattern='[0-9]' message | fields message, no_numbers`

Comando PPL	Descripción	Registros de CloudWatch	Amazon S3	Security Lake	Comando de ejemplo
the section called “sort”	Ordena los resultados mostrados por un nombre de campo. Utilice sort - FieldName para ordenar en orden descendente.	compatible	compatible	compatible	`stats count(), count(`field1`), min(`field1`) as field1Alias, max(`field1`), avg(`field1`) by field2 | sort - field1Alias | head 1000`

Comando PPL	Descripción	Registros de CloudWatch	Amazon S3	Security Lake	Comando de ejemplo
the section called “eval”	Modifica o procesa el valor de un campo y lo almacena en un campo diferente. Esto resulta útil para modificar matemáticamente una columna, aplicar funciones de cadena a una columna o aplicar funciones de fecha a una columna.	compatible	compatible	compatible	`eval field2 = `field1` * 2 | fields field1, field2 | head 20`
the section called “rename”	Cambia el nombre de uno o más campos del resultado de la búsqueda.	compatible	compatible	compatible	`rename field2 as field1 | fields field1`
the section called “head”	Limita los resultados de la consulta a las N primeras filas.	compatible	compatible	compatible	`fields `@message` | head 20`

Comando PPL	Descripción	Registros de CloudWatch	Amazon S3	Security Lake	Comando de ejemplo
the section called “grok”	Analiza un campo de texto con un patrón grok basado en una expresión regular y agrega los resultados al resultado de la búsqueda.	compatible	compatible	compatible	`grok email '.+@%{HOSTNAME:host}' | fields email`
the section called “top”	Busca los valores más frecuentes de un campo.	compatible	compatible	compatible	`top 2 Field1 by Field2`
the section called “dedup”	Elimina entradas duplicadas en función de los campos que especifique.	compatible	compatible	compatible	`dedup field1 | fields field1, field2, field3`
the section called “unirse”	Une dos conjuntos de datos.	compatible	compatible	compatible	`source=customer | join ON c_custkey = o_custkey orders | head 10`

Comando PPL	Descripción	Registros de CloudWatch	Amazon S3	Security Lake	Comando de ejemplo
the section called “lookup”	Enriquece los datos de búsqueda mediante la adición o sustitución de datos de un índice de búsqueda (tabla de dimensiones). Puede ampliar los campos de un índice con valores de una tabla de dimensiones, agregar o reemplazar valores cuando se cumpla la condición de búsqueda.	no compatible	compatible	compatible	`where orderType = 'Cancelled' | lookup account_list, mkt_id AS mkt_code replace amount, account_name as name | stats count(mkt_code), avg(amount) by name`

Comando PPL	Descripción	Registros de CloudWatch	Amazon S3	Security Lake	Comando de ejemplo
the section called “subquery”	Realiza consultas anidadas y complejas en sus instrucciones de lenguaje de procesamiento de canalizaciones (PPL).	compatible	compatible	compatible	`where id in [ subquery source=users | where user in [ subquery source=actions | where action="login" | fields user ] | fields uid ]`
the section called “rare”	Busca los valores menos frecuentes de todos los campos de la lista de campos.	compatible	compatible	compatible	`rare Field1 by Field2`
the section called “trendline”	Calcula los promedios móviles de los campos.	compatible	compatible	compatible	`trendline sma(2, field1) as field1Alias`

Comando PPL	Descripción	Registros de CloudWatch	Amazon S3	Security Lake	Comando de ejemplo
the section called “eventstats”	Enriquece los datos del evento con estadísticas resumidas calculadas. Analiza los campos específicos de sus eventos, calcula varias medidas estadísticas y, a continuación, agrega estos resultados a cada evento original como campos nuevos.	compatible (excepto `count()`)	compatible	compatible	`eventstats sum(field1) by field2`
the section called “flatten”	Aplana un campo. El campo debe ser de este tipo: `struct<?,?> or array<struct<?,?>>`	compatible	compatible	compatible	`source=table | flatten field1`

Comando PPL	Descripción	Registros de CloudWatch	Amazon S3	Security Lake	Comando de ejemplo
the section called “fieldsummary”	Calcula las estadísticas básicas de cada campo (recuento, recuento distinto, mínimo, máximo, promedio, stddev y media).	Compatible (un campo por consulta)	compatible	compatible	`where field1 != 200 | fieldsummary includefields=field1 nulls=true`
the section called “fillnull”	Rellena los campos nulos con el valor que proporcione. Se puede usar en uno o más campos.	compatible	compatible	compatible	`fields field1 | eval field2=field1 | fillnull value=0 field1`
the section called “Expandir”	Divide un campo que contiene varios valores en filas independientes, creando una nueva fila para cada valor del campo especificado.	compatible	compatible	compatible	`expand employee | stats max(salary) as max by state, company`

Comando PPL	Descripción	Registros de CloudWatch	Amazon S3	Security Lake	Comando de ejemplo
the section called “describe”	Obtiene información detallada sobre la estructura y los metadatos de las tablas, los esquemas y los catálogos	no compatible	compatible	compatible	`describe schema.table`

Funciones

Función PPL	Descripción	Registros de CloudWatch	Amazon S3	Security Lake	Comando de ejemplo
the section called “Cadena” (`CONCAT`, `CONCAT_WS` , `LENGTH`, `LOWER`, `LTRIM`, `POSITION`, `REVERSE`, `RIGHT`, `RTRIM`, `SUBSTRING` , `TRIM`, `UPPER`)	Funciones integradas en PPL que pueden manipular y transformar datos de texto y cadenas de texto dentro de las consultas de PPL. Por ejemplo, convertir	compatible	compatible	compatible	`eval col1Len = LENGTH(col1) | fields col1Len`

Función PPL	Descripción	Registros de CloudWatch	Amazon S3	Security Lake	Comando de ejemplo
	mayúsculas y minúsculas, combinar cadenas, extraer partes y limpiar el texto.				

Función PPL	Descripción	Registros de CloudWatch	Amazon S3	Security Lake	Comando de ejemplo
the section called “Fecha y hora” (DAY, DAYOFMONTH , DAY_OF_MONTH ,DAYOFWEEK , DAY_OF_WEEK , DAYOFYEAR , DAY_OF_YEAR , DAYNAME, FROM_UNIXTIME , HOUR, HOUR_OF_DAY , LAST_DAY, LOCALTIMESTAMP , LOCALTIME , MAKE_DATE , MINUTE, MINUTE_OF_HOUR , MONTH, MONTHNAME , MONTH_OF_YEAR , NOW, QUARTER, SECOND, SECOND_OF_MINUTE , SUBDATE, SYSDATE, TIMESTAMP , UNIX_TIMESTAMP , WEEK, WEEKDAY, WEEK_OF_YEAR , DATE_ADD,	Funciones integradas para administrar y transformar los datos de fecha y hora en las consultas PPL. Por ejemplo, date_add, date_format, datediff y current_date.	compatible	compatible	compatible	`eval newDate = ADDDATE(DATE('2020-08-26'), 1) | fields newDate`

Función PPL	Descripción	Registros de CloudWatch	Amazon S3	Security Lake	Comando de ejemplo
DATE_SUB, TIMESTAMP ADD , TIMESTAMP DIFF , UTC_TIMESTAMP , CURRENT_TIMEZONE)					
the section called “Condición” (EXISTS, IF, IFNULL, ISNOTNULL , ISNULL, NULLIF)	Funciones integradas que realizan cálculos en varias filas para generar un único valor resumido. Por ejemplo, sum, count, avg, max y min.	compatible	compatible	compatible	`eval field2 = isnull(col1) | fields field2, col1, field3`

Función PPL	Descripción	Registros de CloudWatch	Amazon S3	Security Lake	Comando de ejemplo
the section called “Math” (ABS, ACOS, ASIN, ATAN, ATAN2, CEIL, CEILING, CONV, COS, COT, CRC32, DEGREES, E, EXP, FLOOR, LN, LOG, LOG2, LOG10, MOD, PI. POW, POWER, RADIANS, RAND, ROUND, SIGN, SIN, SQRT, CBRT)	Funciones integradas para realizar cálculos matemáticos y transformaciones en consultas PPL. Por ejemplo: abs (valor absoluto), round (redondea números), sqrt (raíz cuadrada), pow (cálculo de potencia) y ceil (redondea al entero más cercano).	compatible	compatible	compatible	`eval field2 = ACOS(col1) | fields col1`

Función PPL	Descripción	Registros de CloudWatch	Amazon S3	Security Lake	Comando de ejemplo
the section called “Expresiones” (Operadores aritméticos (`+`, `-`, `*`), operadores de predicados (`>. <`, `IN`))	Las funciones integradas para las expresiones, en particular las expresiones de valores, devuelven un valor escalar. Las expresiones tienen distintos tipos y formas.	compatible	compatible	compatible	`where age > (25 + 5) | fields age`
the section called “Dirección IP” (`CIDRMATCH `)	Funciones integradas para administrar direcciones IP, como el CIDR.	compatible	compatible	compatible	`where cidrmatch (ip, '******** ***/24') | fields ip`

Función PPL	Descripción	Registros de CloudWatch	Amazon S3	Security Lake	Comando de ejemplo
the section called “JSON” (ARRAY_LENGTH , ARRAY_LENGTH , JSON, JSON_ARRAY , JSON_EXTRACT , JSON_KEYS , JSON_OBJECT , JSON_VALID , TO_JSON_STRING)	Funciones integradas para el manejo de JSON, incluidas las matrices, la extracción y la validación.	compatible	compatible	compatible	`eval `json_extract('{"a":"b"}', '$.a')` = json_extract('{"a":"b"}', '$a')`
the section called “Lambda” (EXISTS, FILTER, REDUCE, TRANSFORM)	Funciones integradas para el manejo de JSON, incluidas las matrices, la extracción y la validación.	no compatible	compatible	compatible	`eval array = json_array(1, -1, 2), result = filter(array, x -> x > 0) | fields result`

Función PPL	Descripción	Registros de CloudWatch	Amazon S3	Security Lake	Comando de ejemplo
the section called "Criptográfica" (MD5, SHA1, SHA2)	Funciones integradas que le permiten generar huellas dactilares únicas de los datos, que se pueden utilizar para la verificación, la comparación o como parte de protocolos de seguridad más complejos.	compatible	compatible	compatible	`eval` `MD5('hello')` = MD5('hello') | fields `MD5('hello')`

Información adicional para los usuarios de Información de registros de CloudWatch que utilizan OpenSearch PPL

Aunque Información de registros de CloudWatch admite la mayoría de los comandos y funciones PPL de OpenSearch, algunos comandos y funciones no son compatibles por el momento. Por ejemplo, actualmente no admite los comandos de búsqueda en PPL. A partir del 2 de junio de 2025, Información de registros de CloudWatch ahora admite las funciones JOIN, subconsultas, Flatten,

Fillnull, Expand, Cidrmatch y JSON en PPL. Para obtener una lista completa de los comandos y funciones de consulta compatibles, consulte las columnas de Registros de Amazon CloudWatch en las tablas anteriores.

Consultas y cuotas de ejemplo

Lo siguiente se aplica tanto a los usuarios de Información de registros de CloudWatch como a los usuarios de OpenSearch que consultan datos de CloudWatch.

Para obtener información acerca de los límites que se aplican al consultar Registros de CloudWatch desde OpenSearch Service, consulte Cuotas de Registros de CloudWatch en la Guía del usuario de Registros de Amazon CloudWatch. Los límites se refieren al número de grupos de Registros de CloudWatch que se pueden consultar, el número máximo de consultas simultáneas que se pueden ejecutar, el tiempo máximo de ejecución de las consultas y el número máximo de filas devueltas en los resultados. Los límites son los mismos independientemente del idioma que utilice para consultar Registros de CloudWatch (es decir, OpenSearch PPL, SQL y Logs Insights QL).

Comandos PPL

Temas

- comentario
- Comando de correlación
- dedup command
- Comando describe
- Comando eval
- comando eventstats
- Comando expand
- Comando explain
- Comando fillnull
- Comando fields
- Comando flatten
- Comando grok
- Comando head
- Comando join
- Comando lookup

- Comando parse
- Comando patterns
- Comando rare
- Comando rename
- Comando search
- Comando sort
- Comandos stats
- Comando subquery
- Comando top
- Comando trendline
- Comando where
- Resumen de campo
- Comando expand
- Funciones PPL

comentario

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando PPL, consulte the section called “Comandos”.

PPL admite tanto los comentarios de línea como los comentarios bloqueados. El sistema no evalúa el texto de los comentarios.

Comentarios de línea

Los comentarios de línea comienzan con dos barras diagonales // y terminan con una línea nueva.

Ejemplo:

```
os> source=accounts | top gender // finds most common gender of all the accounts
fetched rows / total rows = 2/2
```

```
+----------+
| gender   |
|----------|
| M        |
| F        |
+----------+
```

Comentarios bloqueados

Los comentarios bloqueados comienzan con una barra seguida de un asterisco * y terminan con un asterisco seguido de una barra */.

Ejemplo:

```
os> source=accounts | dedup 2 gender /* dedup the document with gender field keep 2
 duplication */ | fields account_number, gender
fetched rows / total rows = 3/3
+------------------+----------+
| account_number   | gender   |
|------------------+----------|
| 1                | M        |
| 6                | M        |
| 13               | F        |
+------------------+----------+
```

Comando de correlación

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando PPL, consulte the section called “Comandos”.

Puede correlacionar diferentes orígenes de datos según dimensiones y plazos comunes.

Esta correlación es crucial cuando se trata de grandes cantidades de datos de varios mercados verticales que comparten los mismos periodos de tiempo pero que no están sincronizados formalmente.

Al correlacionar estos diferentes orígenes de datos en función de plazos y dimensiones similares, puede enriquecer sus datos y descubrir información valiosa.

Ejemplo

El dominio de la observabilidad tiene tres orígenes de datos distintos:

- Registros
- Métricas
- Registros de seguimiento

Estas orígenes de datos pueden compartir dimensiones comunes. Para realizar la transición de un origen de datos a otro, debe correlacionarlos correctamente. Mediante las convenciones de nomenclatura semántica, puede identificar los elementos compartidos en los registros, los rastros y las métricas.

Ejemplo:


```
{
  "@timestamp": "2018-07-02T22:23:00.186Z",
  "aws": {
    "elb": {
      "backend": {
        "http": {
          "response": {
            "status_code": 500
          }
        },
        "ip": "********",
        "port": "80"
      },
      ...
     "target_port": [
        "10.0.0.1:80"
      ],
      "target_status_code": [
        "500"
      ],
      "traceId": "Root=1-58337262-36d228ad5d99923122bbe354",
      "type": "http"
    }
  },
  "cloud": {
    "provider": "aws"
  },
```

```
  "http": {
    "request": {
    ...
  },
  "communication": {
    "source": {
      "address": "**************",
      "ip": "**************",
      "port": 2817
    }
  },
  "traceId": "Root=1-58337262-36d228ad5d99923122bbe354"
}
```

En este ejemplo, se muestra un registro de AWS ELB que proviene de un servicio que reside en AWS. Muestra una respuesta backend de HTTP con un código de estado 500, lo que indica un error. Esto podría activar una alerta o formar parte de su proceso de supervisión habitual. El siguiente paso es recopilar datos relevantes sobre este suceso para llevar a cabo una investigación exhaustiva.

Si bien es posible que se sienta tentado a consultar todos los datos relacionados con el periodo de tiempo, este enfoque puede resultar abrumador. Podría terminar con demasiada información y dedicar más tiempo a filtrar los datos irrelevantes que a identificar la raíz del problema.

En su lugar, puede utilizar un enfoque más específico mediante la correlación de los datos de diferentes orígenes. Puede utilizar estas dimensiones para la correlación:

- IP: `"ip": "10.0.0.1" | "ip": "**************"`
- Puerto: `"port": 2817 | "target_port": "10.0.0.1:80"`

Si tiene acceso a índices de métricas y trazas adicionales y está familiarizado con la estructura de su esquema, puede crear una consulta de correlación más precisa.

Este es un ejemplo de un documento de índice de rastreo que contiene información HTTP que quizás quiera correlacionar:

```
{
  "traceId": "c1d985bd02e1dbb85b444011f19a1ecc",
  "spanId": "55a698828fe06a42",
  "traceState": [],
  "parentSpanId": "",
  "name": "mysql",
```

```
  "kind": "CLIENT",
  "@timestamp": "2021-11-13T20:20:39+00:00",
  "events": [
    {
      "@timestamp": "2021-03-25T17:21:03+00:00",
       ...
    }
  ],
  "links": [
    {
      "traceId": "c1d985bd02e1dbb85b444011f19a1ecc",
      "spanId": "55a698828fe06a42w2",
      },
      "droppedAttributesCount": 0
    }
  ],
  "resource": {
    "service@name": "database",
    "telemetry@sdk@name": "opentelemetry",
    "host@hostname": "ip-172-31-10-8.us-west-2.compute.internal"
  },
  "status": {
    ...
  },
  "attributes": {
    "http": {
      "user_agent": {
        "original": "Mozilla/5.0"
      },
      "network": {
         ...
        }
      },
      "request": {
         ...
        }
      },
      "response": {
        "status_code": "200",
        "body": {
          "size": 500
        }
      },
      "client": {
```

```
        "server": {
          "socket": {
            "address": "***********",
            "domain": "example.com",
            "port": 80
          },
          "address": "***********",
          "port": 80
        },
        "resend_count": 0,
        "url": {
          "full": "http://example.com"
        }
      },
      "server": {
        "route": "/index",
        "address": "***********",
        "port": 8080,
        "socket": {
         ...
        },
        "client": {
         ...
         }
        },
        "url": {
         ...
        }
      }
    }
  }
}
```

En este enfoque, puede ver el `traceId` y la `ip` de cliente/servidor de http que se puede correlacionar con los registros de elb para comprender mejor el comportamiento y el estado del sistema.

Nuevo comando de consulta de correlación

Este es el nuevo comando que permitiría este tipo de investigación:

```
source alb_logs, traces | where alb_logs.ip="10.0.0.1" AND
 alb_logs.cloud.provider="aws"|
```

```
correlate exact fields(traceId, ip) scope(@timestamp, 1D) mapping(alb_logs.ip =
 traces.attributes.http.server.address, alb_logs.traceId = traces.traceId )
```

A continuación, se muestra lo que hace cada parte del comando:

1. `source alb_logs, traces`: esto selecciona los orígenes de datos que desea correlacionar.
2. `where ip="10.0.0.1" AND cloud.provider="aws"`: esto reduce el alcance de la búsqueda.
3. `correlate exact fields(traceId, ip)`: esto le indica al sistema que correlacione los datos en función de las coincidencias exactas de los siguientes campos:
   - El campo `ip` tiene una condición de filtro explícita, por lo que se utilizará en la correlación de todas los orígenes de datos.
   - El campo `traceId` no tiene un filtro explícito, por lo que coincidirá con los mismos traceId en todas las orígenes de datos.

Los nombres de los campos indican el significado lógico de la función dentro del comando de correlación. La condición de unión real depende de la sentencia de asignación que proporcione.

El término `exact` significa que las declaraciones de correlación requerirán que todos los campos coincidan para cumplir con la declaración de consulta.

El término `approximate` intentará coincidir en el mejor de los casos y no rechazará las filas con coincidencias parciales.

Tratamiento de diferentes asignaciones de campos

En los casos en que el mismo campo lógico (por ejemplo `ip`) tenga nombres diferentes en los orígenes de datos, debe proporcionar la asignación explícita de los campos de ruta. Para solucionar este problema, puede ampliar las condiciones de correlación para que coincidan con distintos nombres de campo con significados lógicos similares. A continuación le explicamos cómo puede configurar esto:

```
alb_logs.ip = traces.attributes.http.server.address, alb_logs.traceId = traces.traceId
```

Para cada campo que participe en la unión de correlación, debe proporcionar una sentencia de asignación relevante que incluya todas las tablas que se van a unir mediante este comando de correlación.

Ejemplo

En este ejemplo, hay dos orígenes: `alb_logs, traces`

Hay 2 campos: `traceId, ip`

Hay dos instrucciones de asignación: `alb_logs.ip = traces.attributes.http.server.address, alb_logs.traceId = traces.traceId`

Delimitación de los plazos de correlación

Para simplificar el trabajo realizado por el motor de ejecución (controlador), puede agregar la sentencia de alcance. Esto indica explícitamente a la consulta de unión el momento en el que debe realizarse esta búsqueda.

`scope(@timestamp, 1D)` i

En este ejemplo, el ámbito de búsqueda se centra en el día a día, por lo que las correlaciones que aparecen el mismo día se agrupan. Este mecanismo de determinación del alcance simplifica y permite un mejor control de los resultados, lo que permite una resolución de búsqueda incremental en función de sus necesidades.

Controladores de apoyo

El nuevo comando de correlación es en realidad un comando de unión “oculto”. Por lo tanto, solo los siguientes controladores PPL admiten este comando. En estos controladores, el comando de correlación se traducirá directamente al plan lógico Catalyst Join correspondiente.

Ejemplo

```
source alb_logs, traces, metrics | where ip="10.0.0.1" AND
cloud.provider="aws"| correlate exact on (ip, port) scope(@timestamp,
2018-07-02T22:23:00, 1 D)
```

Plan lógico:

```
'Project [*]
+- 'Join Inner, ('ip && 'port)
   :- 'Filter (('ip === "10.0.0.1" & 'cloud.provider === "aws") &
 inTimeScope('@timestamp, "2018-07-02T22:23:00", "1 D"))
      +- 'UnresolvedRelation [alb_logs]
   +- 'Join Inner, ('ip & 'port)
```

```
     :- 'Filter (('ip === "10.0.0.1" & 'cloud.provider === "aws") &
inTimeScope('@timestamp, "2018-07-02T22:23:00", "1 D"))
        +- 'UnresolvedRelation [traces]
     +- 'Filter (('ip === "10.0.0.1" & 'cloud.provider === "aws") &
inTimeScope('@timestamp, "2018-07-02T22:23:00", "1 D"))
        +- 'UnresolvedRelation [metrics]
```

El motor catalizador optimiza esta consulta según el orden de unión más eficiente.

dedup command

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando PPL, consulte the section called “Comandos”.

Utilice el comando `dedup` para eliminar documentos idénticos de los resultados de la búsqueda en función de los campos especificados.

Sintaxis

Utilice la siguiente sintaxis:

```
dedup [int] <field-list> [keepempty=<bool>] [consecutive=<bool>]
```

`int`

- Opcional.
- El comando `dedup` conserva varios eventos para cada combinación cuando se especifica <int>. El número para <int> debe ser mayor que 0. Si no especifica un número, solo se conservara el primer evento que se produzca. Todos los demás duplicados se eliminan de los resultados.
- Valor predeterminado: 1

`keepempty`

- Opcional.
- Si es verdadero, mantiene los documentos en los que algún campo de la lista de campos tenga un valor NULL o MISSING.

- Predeterminado: false

`consecutive`

- Opcional.
- Si es verdadero, elimina solo los eventos con combinaciones de valores duplicados consecutivas.
- Predeterminado: false

`field-list`

- Obligatorio.
- Una lista de campos delimitada por comas. Se requiere como mínimo un campo de metadatos.

Ejemplo 1: Deduplicación por un campo

En este ejemplo se muestra cómo desdoblar documentos mediante el campo de género.

Consulta PPL:

```
os> source=accounts | dedup gender | fields account_number, gender;
fetched rows / total rows = 2/2
+------------------+----------+
| account_number   | gender   |
|------------------+----------|
| 1                | M        |
| 13               | F        |
+------------------+----------+
```

Ejemplo 2: Conserve 2 documentos duplicados

En el ejemplo se muestra cómo deduplicar los documentos con el campo de género, conservando dos duplicados.

Consulta PPL:

```
os> source=accounts | dedup 2 gender | fields account_number, gender;
fetched rows / total rows = 3/3
+------------------+----------+
| account_number   | gender   |
```

```
|------------------+----------|
| 1                | M        |
| 6                | M        |
| 13               | F        |
+------------------+----------+
```

Ejemplo 3: Mantener o ignorar el campo vacío de forma predeterminada

El ejemplo muestra cómo deduplicar el documento manteniendo el campo de valor nulo.

Consulta PPL:

```
os> source=accounts | dedup email keepempty=true | fields account_number, email;
fetched rows / total rows = 4/4
+------------------+-----------------------+
| account_number   | email                 |
+------------------+-----------------------+
| 1                | john_doe@example.com  |
| 6                | jane_doe@example.com  |
| 13               | null                  |
| 18               | juan_li@example.com   |
+------------------+-----------------------+
```

El ejemplo muestra cómo deduplicar el documento ignorando el campo de valor vacío.

Consulta PPL:

```
os> source=accounts | dedup email | fields account_number, email;
fetched rows / total rows = 3/3
+------------------+-----------------------+
| account_number   | email                 |
+------------------+-----------------------+
| 1                | john_doe@example.com  |
| 6                | jane_doe@example.com  |
| 18               | juan_li@example.com   |
+------------------+-----------------------+
```

Ejemplo 4: Deduplicación en documentos consecutivos

En el ejemplo se muestra cómo deduplicar en documentos consecutivos.

Consulta PPL:

```
os> source=accounts | dedup gender consecutive=true | fields account_number, gender;
fetched rows / total rows = 3/3
+------------------+----------+
| account_number   | gender   |
+------------------+----------+
| 1                | M        |
| 13               | F        |
| 18               | M        |
+------------------+----------+
```

Ejemplos adicionales

- `source = table | dedup a | fields a,b,c`
- `source = table | dedup a,b | fields a,b,c`
- `source = table | dedup a keepempty=true | fields a,b,c`
- `source = table | dedup a,b keepempty=true | fields a,b,c`
- `source = table | dedup 1 a | fields a,b,c`
- `source = table | dedup 1 a,b | fields a,b,c`
- `source = table | dedup 1 a keepempty=true | fields a,b,c`
- `source = table | dedup 1 a,b keepempty=true | fields a,b,c`
- `source = table | dedup 2 a | fields a,b,c`
- `source = table | dedup 2 a,b | fields a,b,c`
- `source = table | dedup 2 a keepempty=true | fields a,b,c`
- `source = table | dedup 2 a,b keepempty=true | fields a,b,c`
- `source = table | dedup 1 a consecutive=true| fields a,b,c` (no se admite la deduplicación consecutiva)

Limitación

- En `| dedup 2 a, b keepempty=false`

```
DataFrameDropColumns('_row_number_)
+- Filter ('_row_number_ <= 2) // allowed duplication = 2
   +- Window [row_number() windowspecdefinition('a, 'b, 'a ASC NULLS FIRST, 'b ASC
 NULLS FIRST, specifiedwindowframe(RowFrame, unboundedpreceding$(), currentrow$()))
 AS _row_number_], ['a, 'b], ['a ASC NULLS FIRST, 'b ASC NULLS FIRST]
```

```
      +- Filter (isnotnull('a) AND isnotnull('b)) // keepempty=false
         +- Project
            +- UnresolvedRelation
```

- En `| dedup 2 a, b keepempty=true`

```
Union
:- DataFrameDropColumns('_row_number_)
:  +- Filter ('_row_number_ <= 2)
:     +- Window [row_number() windowspecdefinition('a, 'b, 'a ASC NULLS FIRST, 'b ASC
 NULLS FIRST, specifiedwindowframe(RowFrame, unboundedpreceding$(), currentrow$()))
 AS _row_number_], ['a, 'b], ['a ASC NULLS FIRST, 'b ASC NULLS FIRST]
:        +- Filter (isnotnull('a) AND isnotnull('b))
:           +- Project
:              +- UnresolvedRelation
+- Filter (isnull('a) OR isnull('b))
   +- Project
      +- UnresolvedRelation
```

Comando describe

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando PPL, consulte the section called “Comandos”.

Utilice el comando `describe` para obtener información detallada sobre la estructura y los metadatos de las tablas, los esquemas y los catálogos. A continuación, se muestran varios ejemplos y casos de uso del comando `describe`.

Describe

- `describe table` Este comando es igual al comando `DESCRIBE EXTENDED table` de SQL
- `describe schema.table`
- `describe schema.\`table\``
- `describe catalog.schema.table`
- `describe catalog.schema.\`table\``
- `describe \`catalog\`.\`schema\`.\`table\``

Comando eval

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando PPL, consulte the section called “Comandos”.

El comando `eval` evalúa la expresión y agrega el resultado al resultado de la búsqueda.

Sintaxis

Utilice la siguiente sintaxis:

```
eval <field>=<expression> ["," <field>=<expression> ]...
```

- `field`: obligatorio. Si el nombre de campo no existe, se agrega un campo nuevo. Si el nombre de campo ya existe, se sobrescribirá.
- `expression`: obligatorio. Cualquier expresión admitida por el sistema.

Ejemplo 1: creación de un campo nuevo

En este ejemplo se muestra cómo crear un nuevo campo `doubleAge` para cada documento. El nuevo `doubleAge` es el resultado de la evaluación de la edad multiplicado por 2.

Consulta PPL:

```
os> source=accounts | eval doubleAge = age * 2 | fields age, doubleAge ;
fetched rows / total rows = 4/4
+-------+-------------+
| age   | doubleAge   |
|-------+-------------|
| 32    | 64          |
| 36    | 72          |
| 28    | 56          |
| 33    | 66          |
+-------+-------------+
```

Ejemplo 2: anular el campo existente

En este ejemplo, se muestra cómo anular el campo de edad existente por edad más 1.

Consulta PPL:

```
os> source=accounts | eval age = age + 1 | fields age ;
fetched rows / total rows = 4/4
+-------+
| age   |
|-------|
| 33    |
| 37    |
| 29    |
| 34    |
+-------+
```

Ejemplo 3: cree el nuevo campo con el campo definido en eval

En este ejemplo se muestra cómo crear un nuevo campo `ddAge` con un campo definido en el comando eval. El nuevo campo `ddAge` es el resultado de la evaluación de `doubleAge` multiplicado por 2, donde `doubleAge` se define en el comando eval.

Consulta PPL:

```
os> source=accounts | eval doubleAge = age * 2, ddAge = doubleAge * 2 | fields age,
 doubleAge, ddAge ;
fetched rows / total rows = 4/4
+-------+-------------+---------+
| age   | doubleAge   | ddAge   |
|-------+-------------+---------|
| 32    | 64          | 128     |
| 36    | 72          | 144     |
| 28    | 56          | 112     |
| 33    | 66          | 132     |
+-------+-------------+---------+
```

Supuestos: `a`, `b` y `c` son campos existentes en `table`

Ejemplos adicionales

- `source = table | eval f = 1 | fields a,b,c,f`
- `source = table | eval f = 1` (campos de salida a, b, c, f)
- `source = table | eval n = now() | eval t = unix_timestamp(a) | fields n,t`

- `source = table | eval f = a | where f > 1 | sort f | fields a,b,c | head 5`
- `source = table | eval f = a * 2 | eval h = f * 2 | fields a,f,h`
- `source = table | eval f = a * 2, h = f * 2 | fields a,f,h`
- `source = table | eval f = a * 2, h = b | stats avg(f) by h`
- `source = table | eval f = ispresent(a)`
- `source = table | eval r = coalesce(a, b, c) | fields r`
- `source = table | eval e = isempty(a) | fields e`
- `source = table | eval e = isblank(a) | fields e`
- `source = table | eval f = case(a = 0, 'zero', a = 1, 'one', a = 2, 'two', a = 3, 'three', a = 4, 'four', a = 5, 'five', a = 6, 'six', a = 7, 'se7en', a = 8, 'eight', a = 9, 'nine')`
- `source = table | eval f = case(a = 0, 'zero', a = 1, 'one' else 'unknown')`
- `source = table | eval f = case(a = 0, 'zero', a = 1, 'one' else concat(a, ' is an incorrect binary digit'))`
- `source = table | eval f = a in ('foo', 'bar') | fields f`
- `source = table | eval f = a not in ('foo', 'bar') | fields f`

Evaluación con un ejemplo de caso:

```
source = table | eval e = eval status_category =
case(a >= 200 AND a < 300, 'Success',
a >= 300 AND a < 400, 'Redirection',
a >= 400 AND a < 500, 'Client Error',
a >= 500, 'Server Error'
else 'Unknown')
```

Evaluación con otro ejemplo de caso:

Supuestos: a, b y c son campos existentes en `table`

Ejemplos adicionales

- `source = table | eval f = 1 | fields a,b,c,f`

- `source = table | eval f = 1` (campos de salida a, b, c, f)
- `source = table | eval n = now() | eval t = unix_timestamp(a) | fields n,t`
- `source = table | eval f = a | where f > 1 | sort f | fields a,b,c | head 5`
- `source = table | eval f = a * 2 | eval h = f * 2 | fields a,f,h`
- `source = table | eval f = a * 2, h = f * 2 | fields a,f,h`
- `source = table | eval f = a * 2, h = b | stats avg(f) by h`
- `source = table | eval f = ispresent(a)`
- `source = table | eval r = coalesce(a, b, c) | fields r`
- `source = table | eval e = isempty(a) | fields e`
- `source = table | eval e = isblank(a) | fields e`
- `source = table | eval f = case(a = 0, 'zero', a = 1, 'one', a = 2, 'two', a = 3, 'three', a = 4, 'four', a = 5, 'five', a = 6, 'six', a = 7, 'se7en', a = 8, 'eight', a = 9, 'nine')`
- `source = table | eval f = case(a = 0, 'zero', a = 1, 'one' else 'unknown')`
- `source = table | eval f = case(a = 0, 'zero', a = 1, 'one' else concat(a, ' is an incorrect binary digit'))`
- `source = table | eval f = a in ('foo', 'bar') | fields f`
- `source = table | eval f = a not in ('foo', 'bar') | fields f`

Evaluación con un ejemplo de caso:

```
source = table | eval e = eval status_category =
case(a >= 200 AND a < 300, 'Success',
a >= 300 AND a < 400, 'Redirection',
a >= 400 AND a < 500, 'Client Error',
a >= 500, 'Server Error'
else 'Unknown')
```

Evaluación con otro ejemplo de caso:

```
source = table |  where ispresent(a) |
```

```
eval status_category =
 case(a >= 200 AND a < 300, 'Success',
  a >= 300 AND a < 400, 'Redirection',
  a >= 400 AND a < 500, 'Client Error',
  a >= 500, 'Server Error'
  else 'Incorrect HTTP status code'
 )
 | stats count() by status_category
```

Limitaciones

- No se admite el anular los campos existentes. Las consultas que lo intenten generarán excepciones con el mensaje “La referencia 'a' es ambigua”.

```
- `source = table | eval a = 10 | fields a,b,c`
- `source = table | eval a = a * 2 | stats avg(a)`
- `source = table | eval a = abs(a) | where a > 0`
- `source = table | eval a = signum(a) | where a < 0`
```

comando eventstats

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando PPL, consulte the section called “Comandos”.

Utilice el comando `eventstats` para enriquecer los datos del evento con estadísticas resumidas calculadas. Funciona mediante el análisis de campos específicos de los eventos, el cálculo de diversas medidas estadísticas y, a continuación, la adición de estos resultados como campos nuevos a cada evento original.

Aspectos clave de eventstats

1. Realiza cálculos en todo el conjunto de resultados o dentro de grupos definidos.
2. Los eventos originales permanecen intactos y se agregan nuevos campos para contener los resultados estadísticos.
3. El comando es particularmente útil para realizar análisis comparativos, identificar valores atípicos o proporcionar contexto adicional a eventos individuales.

Diferencia entre stats y eventstats

Los comandos `stats` y `eventstats` se utilizan para calcular estadísticas, pero tienen algunas diferencias clave en su funcionamiento y en lo que producen.

Formato de salida

- `stats`: produce una tabla de resumen con solo las estadísticas calculadas.
- `eventstats`: agrega las estadísticas calculadas como campos nuevos a los eventos existentes y conserva los datos originales.

Retención de eventos

- `stats`: reduce el conjunto de resultados a solo un resumen estadístico y descarta los eventos individuales.
- `eventstats`: retiene todos los eventos originales y agrega nuevos campos con las estadísticas calculadas.

Casos de uso

- `stats`: ideal para crear paneles o informes resumidos. Suele utilizarse como comando final para resumir los resultados.
- `eventstats`: útil cuando se necesita enriquecer los eventos con un contexto estadístico para su posterior análisis o filtrado. Se puede utilizar en mitad de una búsqueda para agregar estadísticas que se pueden utilizar en comandos posteriores.

Sintaxis

Utilice la siguiente sintaxis:

```
eventstats <aggregation>... [by-clause]
```

agregación

- Obligatorio.
- Una función de agregación.
- El argumento de agregación debe ser un campo.

by-clause

- Opcional.
- Sintaxis: `by [span-expression,] [field,]...`
- La cláusula by puede incluir campos y expresiones, como funciones escalares y funciones de agregación. También puede usar la cláusula span para dividir un campo específico en grupos de intervalos iguales. A continuación, el comando eventstats realiza la agregación en función de estos buckets de intervalo.
- Predeterminado: si no especificas una cláusula by, el comando eventstats agrega todo el conjunto de resultados.

span-expression

- Opcional, como máximo uno.
- Sintaxis: `span(field_expr, interval_expr)`
- La unidad de la expresión de intervalo es la unidad natural por defecto. Sin embargo, para los campos de tipo fecha y hora, debe especificar la unidad en la expresión de intervalo cuando utilice unidades de fecha y hora.

  Por ejemplo, para dividir el campo `age` en buckets por 10 años, utilice `span(age, 10)`. Para los campos basados en el tiempo, puede dividir un campo `timestamp` en intervalos de una hora si utiliza `span(timestamp, 1h)`.

Unidades de tiempo disponibles

Unidades de intervalo
milisegundo (ms)
segundo (s)
minuto (m, distingue entre mayúsculas y minúsculas)
hora (h)
día (d)
semana (s)

Unidades de intervalo
mes (M, distingue entre mayúsculas y minúsculas)
trimestre (q)
año (y)

Funciones de agregación

COUNT

COUNT devuelve un recuento del número de expresiones en las filas recuperadas por una sentencia SELECT.

No se admiten consultas de uso COUNT en Registros de CloudWatch.

Ejemplo:

```
os> source=accounts | eventstats count();
fetched rows / total rows = 4/4
+----------------+----------+-----------+----------+-----+--------+--------------------
+------------+--------------------------+--------+-------+---------+
| account_number | balance  | firstname | lastname | age | gender | address
 | employer   | email                     | city   | state | count() |
+----------------+----------+-----------+----------+-----+--------+--------------------
+------------+--------------------------+--------+-------+---------+
| 1              | 39225    | Jane      | Doe      | 32  | M      | *** Any Lane
 | AnyCorp    | janedoe@anycorp.com       | Brogan | IL    | 4       |
| 6              | 5686     | Mary      | Major    | 36  | M      | 671 Example Street
 | AnyCompany | marymajor@anycompany.com | Dante  | TN    | 4       |
| 13             | 32838    | Nikki     | Wolf     | 28  | F      | 789 Any Street
 | AnyOrg     |                           | Nogal  | VA    | 4       |
| 18             | 4180     | Juan      | Li       | 33  | M      | *** Example Court
 |            | juanli@exampleorg.com     | Orick  | MD    | 4       |
+----------------+----------+-----------+----------+-----+--------+--------------------
+------------+--------------------------+--------+-------+---------+
```

SUM

SUM(expr) devuelve la suma de expr.

Ejemplo:

```
os> source=accounts | eventstats sum(age) by gender;
fetched rows / total rows = 4/4
+----------------+----------+-----------+----------+-----+--------
+-----------------------+------------+--------------------------+--------+-------
+--------------------+
| account_number | balance  | firstname | lastname | age | gender | address
   | employer   | email                    | city   | state | sum(age) by gender |
+----------------+----------+-----------+----------+-----+--------
+-----------------------+------------+--------------------------+--------+-------
+--------------------+
| 1              | 39225    | Jane      | Doe      | 32  | M      | 880 Any Lane
   | AnyCorp    | janedoe@anycorp.com      | Brogan | IL    | 101                |
| 6              | 5686     | Mary      | Major    | 36  | M      | 671 Example Street
   | AnyCompany | marymajor@anycompany.com | Dante  | TN    | 101                |
| 13             | 32838    | Nikki     | Wolf     | 28  | F      | 789 Any Street
   | AnyOrg     |                          | Nogal  | VA    | 28                 |
| 18             | 4180     | Juan      | Li       | 33  | M      | 467 Example Court
   |            | juanli@exampleorg.com    | Orick  | MD    | 101                |
+----------------+----------+-----------+----------+-----+--------
+-----------------------+------------+--------------------------+--------+-------
+--------------------+
```

AVG

`AVG(expr)` devuelve el valor medio de expr.

Ejemplo:

```
os> source=accounts | eventstats avg(age) by gender;
fetched rows / total rows = 4/4
+----------------+----------+-----------+----------+-----+--------
+-----------------------+------------+--------------------------+--------+-------
+--------------------+
| account_number | balance  | firstname | lastname | age | gender | address
   | employer    | email                    | city   | state | avg(age) by gender |
+----------------+----------+-----------+----------+-----+--------
+-----------------------+------------+--------------------------+--------+-------
+--------------------+
| 1              | 39225    | Jane      | Doe      | 32  | M      | 880 Any Lane
   | AnyCorp     | janedoe@anycorp.com      | Brogan | IL    | 33.67              |
```

```
| 6              | 5686     | Mary      | Major    | 36  | M      | 671 Example Street
   | Any Company | marymajor@anycompany.com | Dante  | TN    | 33.67             |
| 13             | 32838    | Nikki     | Wolf     | 28  | F      | 789 Any Street
   | AnyOrg      |                          | Nogal  | VA    | 28.00             |
| 18             | 4180     | Juan      | Li       | 33  | M      | 467 Example Court
   |             | juanli@exampleorg.com    | Orick  | MD    | 33.67             |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+--------------------+
```

MAX

`MAX(expr)` devuelve el valor máximo de expr.

Ejemplo

```
os> source=accounts | eventstats max(age);
fetched rows / total rows = 4/4
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----------+
| account_number | balance  | firstname | lastname | age | gender | address
   | employer    | email                    | city   | state | max(age)  |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----------+
| 1              | 39225    | Jane      | Doe      | 32  | M      | 880 Any Lane
   | AnyCorp     | janedoe@anycorp.com      | Brogan | IL    | 36        |
| 6              | 5686     | Mary      | Major    | 36  | M      | 671 Example Street
   | Any Company | marymajor@anycompany.com | Dante  | TN    | 36        |
| 13             | 32838    | Nikki     | Wolf     | 28  | F      | 789 Any Street
   | AnyOrg      |                          | Nogal  | VA    | 36        |
| 18             | 4180     | Juan      | Li       | 33  | M      | *** Example Court
   |             | juanli@exampleorg.com    | Orick  | MD    | 36        |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----------+
```

MIN

`MIN(expr)` devuelve el valor mínimo de expr.

Ejemplo

```
os> source=accounts | eventstats min(age);
fetched rows / total rows = 4/4
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----------+
| account_number | balance  | firstname | lastname | age | gender | address
   | employer    | email                    | city   | state | min(age)  |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----------+
| 1              | 39225    | Jane      | Doe      | 32  | M      | 880 Any Lane
   | AnyCorp     | janedoe@anycorp.com      | Brogan | IL    | 28        |
| 6              | 5686     | Mary      | Major    | 36  | M      | 671 Example Street
   | Any Company | marymajor@anycompany.com | Dante  | TN    | 28        |
| 13             | 32838    | Nikki     | Wolf     | 28  | F      | *** Any Street
   | AnyOrg      |                          | Nogal  | VA    | 28        |
| 18             | 4180     | Juan      | Li       | 33  | M      | *** Example Court
   |             | juanli@exampleorg.com    | Orick  | MD    | 28        |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----------+
```

STDDEV_SAMP

`STDDEV_SAMP(expr)` devuelve la desviación estándar muestral de expr.

Ejemplo

```
os> source=accounts | eventstats stddev_samp(age);
fetched rows / total rows = 4/4
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+------------------------+
| account_number | balance  | firstname | lastname | age | gender | address
   | employer    | email                    | city   | state | stddev_samp(age)       |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+------------------------+
| 1              | 39225    | Jane      | Doe      | 32  | M      | *** Any Lane
   | AnyCorp     | janedoe@anycorp.com      | Brogan | IL    | 3.304037933599835      |
| 6              | 5686     | Mary      | Major    | 36  | M      | 671 Example Street
   | Any Company | marymajor@anycompany.com | Dante  | TN    | 3.304037933599835      |
```

```
| 13             | 32838    | Nikki     | Wolf     | 28  | F      | 789 Any Street
   | AnyOrg      |                          | Nogal  | VA    | 3.304037933599835      |
| 18             | 4180     | Juan      | Li       | 33  | M      | 467 Example Court
   |             | juanli@exampleorg.com    | Orick  | MD    | 3.304037933599835      |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+------------------------+
```

STDDEV_POP

`STDDEV_POP(expr)` devuelve la desviación estándar poblacional de expr.

Ejemplo

```
os> source=accounts | eventstats stddev_pop(age);
fetched rows / total rows = 4/4
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+------------------------+
| account_number | balance  | firstname | lastname | age | gender | address
   | employer    | email                    | city   | state | stddev_pop(age)        |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+------------------------+
| 1              | 39225    | Jane      | Doe      | 32  | M      | 880 Any Lane
   | AnyCorp     | janedoe@anycorp.com      | Brogan | IL    | 2.****************     |
| 6              | 5686     | Mary      | Major    | 36  | M      | *** Example Street
   | Any Company | marymajor@anycompany.com | Dante  | TN    | 2.****************     |
| 13             | 32838    | Nikki     | Wolf     | 28  | F      | *** Any Street
   | AnyOrg      |                          | Nogal  | VA    | 2.****************     |
| 18             | 4180     | Juan      | Li       | 33  | M      | *** Example Court
   |             | juanli@exampleorg.com    | Orick  | MD    | 2.****************     |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+------------------------+
```

PERCENTILE o PERCENTILE_APPROX

`PERCENTILE(expr, percent)` o `PERCENTILE_APPROX(expr, percent)` devuelve el valor percentil aproximado de expr en el porcentaje especificado.

porcentaje

- El número debe ser una constante entre 0 y 100.

Ejemplo

```
os> source=accounts | eventstats percentile(age, 90) by gender;
fetched rows / total rows = 4/4
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+--------------------------------+
| account_number | balance  | firstname | lastname | age | gender | address
    | employer    | email                    | city   | state | percentile(age, 90) by
 gender  |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+--------------------------------+
| 1              | 39225    | Jane      | Doe      | 32  | M      | *** Any Lane
   | AnyCorp     | janedoe@anycorp.com      | Brogan | IL    | 36
       |
| 6              | 5686     | Mary      | Major    | 36  | M      | 671 Example Street
   | Any Company | marymajor@anycompany.com | Dante  | TN    | 36
       |
| 13             | 32838    | Nikki     | Wolf     | 28  | F      | 789 Any Street
   | AnyOrg      |                          | Nogal  | VA    | 28
       |
| 18             | 4180     | Juan      | Li       | 33  | M      | *** Example Court
   |             | juanli@exampleorg.com    | Orick  | MD    | 36
       |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+--------------------------------+
```

Ejemplo 1: calcular el promedio, la suma y el recuento de un campo por grupo

En el ejemplo se muestra cómo calcular la edad media, la edad sumada y el recuento de eventos de todas las cuentas agrupadas por género.

```
os> source=accounts | eventstats avg(age) as avg_age, sum(age) as sum_age, count() as
 count by gender;
fetched rows / total rows = 4/4
```

```
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----------+-----------+--------+
| account_number | balance  | firstname | lastname | age | gender | address
    | employer    | email                    | city   | state | avg_age   | sum_age   |
 count |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----------+-----------+--------+
| 1              | 39225    | Jane      | Doe      | 32  | M      | *** Any Lane
    | AnyCorp     | janedoe@anycorp.com      | Brogan | IL    | 33.666667 | 101       |
 3     |
| 6              | 5686     | Mary      | Major    | 36  | M      | 671 Example Street
    | Any Company | marymajor@anycompany.com | Dante  | TN    | 33.666667 | 101       |
 3     |
| 13             | 32838    | Nikki     | Wolf     | 28  | F      | 789 Any Street
    | AnyOrg      |                          | Nogal  | VA    | 28.000000 | 28        |
 1     |
| 18             | 4180     | Juan      | Li       | 33  | M      | *** Example Court
    |             | juanli@exampleorg.com    | Orick  | MD    | 33.666667 | 101       |
 3     |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----------+-----------+--------+
```

Ejemplo 2: Calcule el recuento en un intervalo

El ejemplo obtiene el recuento de la edad en un intervalo de 10 años.

```
os> source=accounts | eventstats count(age) by span(age, 10) as age_span
fetched rows / total rows = 4/4
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----------+
| account_number | balance  | firstname | lastname | age | gender | address
   | employer    | email                    | city   | state | age_span |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----------+
| 1              | 39225    | Jane      | Doe      | 32  | M      | *** Any Lane
   | AnyCorp     | janedoe@anycorp.com      | Brogan | IL    | 3        |
| 6              | 5686     | Mary      | Major    | 36  | M      | 671 Example Street
   | Any Company | marymajor@anycompany.com | Dante  | TN    | 3        |
```

```
| 13             | 32838    | Nikki     | Wolf     | 28  | F      | 789 Any Street
   | AnyOrg      |                          | Nogal  | VA    | 1        |
| 18             | 4180     | Juan      | Li       | 33  | M      | *** Example Court
   |             | juanli@exampleorg.com    | Orick  | MD    | 3        |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+----------+
```

Ejemplo 3: Calcule el recuento por género y rango

El ejemplo obtiene el recuento de la edad por un intervalo de 5 años y el grupo por género.

```
os> source=accounts | eventstats count() as cnt by span(age, 5) as age_span, gender
fetched rows / total rows = 4/4
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----+
| account_number | balance  | firstname | lastname | age | gender | address
   | employer    | email                    | city   | state | cnt |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----+
| 1              | 39225    | Jane      | Doe      | 32  | M      | *** Any Lane
   | AnyCorp     | janedoe@anycorp.com      | Brogan | IL    | 2   |
| 6              | 5686     | Mary      | Majo     | 36  | M      | 671 Example Street
   | Any Company | hattiebond@anycompany.com | Dante  | TN    | 1   |
| 13             | 32838    | Nikki     | Wolf     | 28  | F      | *** Any Street
   | AnyOrg      |                          | Nogal  | VA    | 1   |
| 18             | 4180     | Juan      | Li       | 33  | M      | *** Example Court
   |             | juanli@exampleorg.com    | Orick  | MD    | 2   |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----+
```

Uso

- `source = table | eventstats avg(a)`
- `source = table | where a < 50 | eventstats avg(c)`
- `source = table | eventstats max(c) by b`
- `source = table | eventstats count(c) by b | head 5`
- `source = table | eventstats distinct_count(c)`

- `source = table | eventstats stddev_samp(c)`
- `source = table | eventstats stddev_pop(c)`
- `source = table | eventstats percentile(c, 90)`
- `source = table | eventstats percentile_approx(c, 99)`

Agregaciones con span

- `source = table | eventstats count(a) by span(a, 10) as a_span`
- `source = table | eventstats sum(age) by span(age, 5) as age_span | head 2`
- `source = table | eventstats avg(age) by span(age, 20) as age_span, country | sort - age_span | head 2`

Agregaciones con intervalo de ventana de tiempo (función de ventana invertida)

- `source = table | eventstats sum(productsAmount) by span(transactionDate, 1d) as age_date | sort age_date`
- `source = table | eventstats sum(productsAmount) by span(transactionDate, 1w) as age_date, productId`

Las agregaciones se agrupan en varios niveles

- `source = table | eventstats avg(age) as avg_state_age by country, state | eventstats avg(avg_state_age) as avg_country_age by country`
- `source = table | eventstats avg(age) as avg_city_age by country, state, city | eval new_avg_city_age = avg_city_age - 1 | eventstats avg(new_avg_city_age) as avg_state_age by country, state | where avg_state_age > 18 | eventstats avg(avg_state_age) as avg_adult_country_age by country`

Comando expand

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando PPL, consulte the section called “Comandos”.

Utilice el comando `expand` para aplanar un campo del tipo:

- `Array<Any>`
- `Map<Any>`

Sintaxis

Utilice la siguiente sintaxis:

```
expand <field> [As alias]
```

campo

- El campo que se va a expandir (explotar). Debe ser de un tipo compatible.

alias

- Opcional. El nombre que se va a utilizar en lugar del nombre del campo original.

Uso

El comando `expand` genera una fila para cada elemento de la matriz o campo de mapa especificado, donde:

- Los elementos de la matriz se convierten en filas individuales.
- Los pares clave-valor de la asignación se dividen en filas independientes y cada valor clave se representa como una fila.

- Cuando se proporciona un alias, los valores desglosados se representan con el alias en lugar del nombre de campo original.

- Esto se puede usar en combinación con otros comandos, como `stats`, `eval` y `parse` para manipular o extraer datos después de la expansión.

Ejemplos

- `source = table | expand employee | stats max(salary) as max by state, company`
- `source = table | expand employee as worker | stats max(salary) as max by state, company`
- `source = table | expand employee as worker | eval bonus = salary * 3 | fields worker, bonus`
- `source = table | expand employee | parse description '(?<email>.+@.+)' | fields employee, email`
- `source = table | eval array=json_array(1, 2, 3) | expand array as uid | fields name, occupation, uid`
- `source = table | expand multi_valueA as multiA | expand multi_valueB as multiB`

Comando explain

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando PPL, consulte the section called “Comandos”.

El comando `explain` lo ayuda a entender los planes de ejecución de las consultas, lo que le permite analizarlas y optimizarlas para obtener un mejor rendimiento. Esta introducción proporciona una descripción general y concisa del propósito del comando explain y de su importancia en la optimización de las consultas.

Comentario

- `source=accounts | top gender // finds most common gender of all the accounts` (comentario de línea)

- `source=accounts | dedup 2 gender /* dedup the document with gender field keep 2 duplication */ | fields account_number, gender` (comentario bloqueado)

Describe

- `describe table` Este comando es igual al comando `DESCRIBE EXTENDED table` de SQL
- `describe schema.table`
- `describe schema.\`table\``
- `describe catalog.schema.table`
- `describe catalog.schema.\`table\``
- `describe \`catalog\`.\`schema\`.\`table\``

Explain

- `explain simple | source = table | where a = 1 | fields a,b,c`
- `explain extended | source = table`
- `explain codegen | source = table | dedup a | fields a,b,c`
- `explain cost | source = table | sort a | fields a,b,c`
- `explain formatted | source = table | fields - a`
- `explain simple | describe table`

Campos

- `source = table`
- `source = table | fields a,b,c`
- `source = table | fields + a,b,c`
- `source = table | fields - b,c`
- `source = table | eval b1 = b | fields - b1,c`

Resumen de campo

- `source = t | fieldsummary includefields=status_code nulls=false`

- `source = t | fieldsummary includefields= id, status_code, request_path nulls=true`
- `source = t | where status_code != 200 | fieldsummary includefields= status_code nulls=true`

Campo anidado

- `source = catalog.schema.table1, catalog.schema.table2 | fields A.nested1, B.nested1`
- `source = catalog.table | where struct_col2.field1.subfield > 'valueA' | sort int_col | fields int_col, struct_col.field1.subfield, struct_col2.field1.subfield`
- `source = catalog.schema.table | where struct_col2.field1.subfield > 'valueA' | sort int_col | fields int_col, struct_col.field1.subfield, struct_col2.field1.subfield`

Filtros

- `source = table | where a = 1 | fields a,b,c`
- `source = table | where a >= 1 | fields a,b,c`
- `source = table | where a < 1 | fields a,b,c`
- `source = table | where b != 'test' | fields a,b,c`
- `source = table | where c = 'test' | fields a,b,c | head 3`
- `source = table | where ispresent(b)`
- `source = table | where isnull(coalesce(a, b)) | fields a,b,c | head 3`
- `source = table | where isempty(a)`
- `source = table | where isblank(a)`
- `source = table | where case(length(a) > 6, 'True' else 'False') = 'True'`
- `source = table | where a not in (1, 2, 3) | fields a,b,c`
- `source = table | where a between 1 and 4`, Nota: Esto devuelve un >= 1 y un <= 4, es decir, [1, 4]
- `source = table | where b not between '2024-09-10' and '2025-09-10'`, Nota: Esto devuelve b >= '**********' y b <= '2025-09-10'

- `source = table | where cidrmatch(ip, '***********/24')`
- `source = table | where cidrmatch(ipv6, '2003:db8::/32')`
- `source = table | trendline sma(2, temperature) as temp_trend`

Consultas relacionadas con la IP

- `source = table | where cidrmatch(ip, '**************')`
- `source = table | where isV6 = false and isValid = true and cidrmatch(ipAddress, '**************')`
- `source = table | where isV6 = true | eval inRange = case(cidrmatch(ipAddress, '2003:***::/32'), 'in' else 'out') | fields ip, inRange`

Filtros complejos

```
source = table | eval status_category =
case(a >= 200 AND a < 300, 'Success',
    a >= 300 AND a < 400, 'Redirection',
    a >= 400 AND a < 500, 'Client Error',
    a >= 500, 'Server Error'
else 'Incorrect HTTP status code')
| where case(a >= 200 AND a < 300, 'Success',
    a >= 300 AND a < 400, 'Redirection',
    a >= 400 AND a < 500, 'Client Error',
    a >= 500, 'Server Error'
else 'Incorrect HTTP status code'
) = 'Incorrect HTTP status code'
```

```
source = table
| eval factor = case(a > 15, a - 14, isnull(b), a - 7, a < 3, a + 1 else 1)
| where case(factor = 2, 'even', factor = 4, 'even', factor = 6, 'even', factor = 8,
 'even' else 'odd') = 'even'
| stats count() by factor
```

Filtros con condiciones lógicas

- `source = table | where c = 'test' AND a = 1 | fields a,b,c`

- `source = table | where c != 'test' OR a > 1 | fields a,b,c | head 1`
- `source = table | where c = 'test' NOT a > 1 | fields a,b,c`

Eval

Supuestos: a, b, c son campos existentes en `table`

- `source = table | eval f = 1 | fields a,b,c,f`
- `source = table | eval f = 1` (campos de salida a, b, c, f)
- `source = table | eval n = now() | eval t = unix_timestamp(a) | fields n,t`
- `source = table | eval f = a | where f > 1 | sort f | fields a,b,c | head 5`
- `source = table | eval f = a * 2 | eval h = f * 2 | fields a,f,h`
- `source = table | eval f = a * 2, h = f * 2 | fields a,f,h`
- `source = table | eval f = a * 2, h = b | stats avg(f) by h`
- `source = table | eval f = ispresent(a)`
- `source = table | eval r = coalesce(a, b, c) | fields r`
- `source = table | eval e = isempty(a) | fields e`
- `source = table | eval e = isblank(a) | fields e`
- `source = table | eval f = case(a = 0, 'zero', a = 1, 'one', a = 2, 'two', a = 3, 'three', a = 4, 'four', a = 5, 'five', a = 6, 'six', a = 7, 'se7en', a = 8, 'eight', a = 9, 'nine')`
- `source = table | eval f = case(a = 0, 'zero', a = 1, 'one' else 'unknown')`
- `source = table | eval f = case(a = 0, 'zero', a = 1, 'one' else concat(a, ' is an incorrect binary digit'))`
- `source = table | eval digest = md5(fieldName) | fields digest`
- `source = table | eval digest = sha1(fieldName) | fields digest`
- `source = table | eval digest = sha2(fieldName,256) | fields digest`
- `source = table | eval digest = sha2(fieldName,512) | fields digest`

Comando fillnull

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando PPL, consulte the section called “Comandos”.

Descripción

Use el comando `fillnull` para reemplazar los valores nulos por un valor especificado en uno o más campos de los resultados de la búsqueda.

Sintaxis

Utilice la siguiente sintaxis:

```
fillnull [with <null-replacement> in <nullable-field>["," <nullable-field>]] | [using
 <source-field> = <null-replacement> [","<source-field> = <null-replacement>]]
```

- null-replacement: obligatorio. El valor utilizado para reemplazar los valores nulos.
- nullable-field: obligatorio. Referencia de campos. Los valores nulos de este campo se sustituirán por el valor especificado en la sustitución de valores nulos.

Ejemplo 1: Fillnull en un campo

En el ejemplo se muestra cómo utilizar fillnull en un solo campo:

```
os> source=logs | fields status_code | eval input=status_code | fillnull with 0 in
 status_code;
| input | status_code |
|-------|-------------|
| 403   | 403         |
| 403   | 403         |
| NULL  | 0           |
| NULL  | 0           |
| 200   | 200         |
| 404   | 404         |
| 500   | 500         |
| NULL  | 0           |
| 500   | 500         |
```

```
| 404   | 404         |
| 200   | 200         |
| 500   | 500         |
| NULL  | 0           |
| NULL  | 0           |
| 404   | 404         |
```

Ejemplo 2: Fillnull aplicado a varios campos

El ejemplo muestra la aplicación fillnull a varios campos.

```
os> source=logs | fields request_path, timestamp | eval
 input_request_path=request_path, input_timestamp = timestamp | fillnull with '???' in
 request_path, timestamp;
| input_request_path | input_timestamp       | request_path | timestamp              |
|--------------------------------------------------------------------------------------|
| /contact           | NULL                  | /contact     | ???                    |
| /home              | NULL                  | /home        | ???                    |
| /about             | 2023-10-01 10:30:00   | /about       | 2023-10-01 10:30:00    |
| /home              | 2023-10-01 10:15:00   | /home        | 2023-10-01 10:15:00    |
| NULL               | 2023-10-01 10:20:00   | ???          | 2023-10-01 10:20:00    |
| NULL               | 2023-10-01 11:05:00   | ???          | 2023-10-01 11:05:00    |
| /about             | NULL                  | /about       | ???                    |
| /home              | 2023-10-01 10:00:00   | /home        | 2023-10-01 10:00:00    |
| /contact           | NULL                  | /contact     | ???                    |
| NULL               | 2023-10-01 10:05:00   | ???          | 2023-10-01 10:05:00    |
| NULL               | 2023-10-01 10:50:00   | ???          | 2023-10-01 10:50:00    |
| /services          | NULL                  | /services    | ???                    |
| /home              | 2023-10-01 10:45:00   | /home        | 2023-10-01 10:45:00    |
| /services          | 2023-10-01 11:00:00   | /services    | 2023-10-01 11:00:00    |
| NULL               | 2023-10-01 10:35:00   | ???          | 2023-10-01 10:35:00    |
```

Ejemplo 3: Se aplica fillnull a varios campos con varios valores de reemplazo nulos.

El ejemplo muestra fillnull con varios valores que se utilizan para reemplazar los valores nulos.

- `/error` en el campo `request_path`
- `1970-01-01 00:00:00` en el campo `timestamp`

```
os> source=logs | fields request_path, timestamp | eval
 input_request_path=request_path, input_timestamp = timestamp | fillnull using
 request_path = '/error', timestamp='1970-01-01 00:00:00';
```

```
| input_request_path | input_timestamp       | request_path | timestamp             |
|-------------------------------------------------------------------------------|
| /contact           | NULL                  | /contact     | 1970-01-01 00:00:00   |
| /home              | NULL                  | /home        | 1970-01-01 00:00:00   |
| /about             | 2023-10-01 10:30:00   | /about       | 2023-10-01 10:30:00   |
| /home              | 2023-10-01 10:15:00   | /home        | 2023-10-01 10:15:00   |
| NULL               | 2023-10-01 10:20:00   | /error       | 2023-10-01 10:20:00   |
| NULL               | 2023-10-01 11:05:00   | /error       | 2023-10-01 11:05:00   |
| /about             | NULL                  | /about       | 1970-01-01 00:00:00   |
| /home              | 2023-10-01 10:00:00   | /home        | 2023-10-01 10:00:00   |
| /contact           | NULL                  | /contact     | 1970-01-01 00:00:00   |
| NULL               | 2023-10-01 10:05:00   | /error       | 2023-10-01 10:05:00   |
| NULL               | 2023-10-01 10:50:00   | /error       | 2023-10-01 10:50:00   |
| /services          | NULL                  | /services    | 1970-01-01 00:00:00   |
| /home              | 2023-10-01 10:45:00   | /home        | 2023-10-01 10:45:00   |
| /services          | 2023-10-01 11:00:00   | /services    | 2023-10-01 11:00:00   |
| NULL               | 2023-10-01 10:35:00   | /error       | 2023-10-01 10:35:00   |
```

Comando fields

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando PPL, consulte the section called “Comandos”.

Utilice el comando `fields` para conservar o eliminar campos del resultado de la búsqueda.

Sintaxis

Utilice la siguiente sintaxis:

```
field [+|-] <field-list>
```

- `index`: opcional.

  Si se utiliza el signo más (+), solo se conservarán los campos especificados en la lista de campos.

  Si se utiliza el signo menos (-), se eliminarán todos los campos especificados en la lista de campos.

Valor predeterminado: +

- `field list`: obligatorio. Una lista delimitada por comas de campos para conservar o eliminar.

Ejemplo 1: seleccionar campos específicos del resultado

En este ejemplo se muestra cómo obtener campos `account_number`, `firstname` y `lastname` de los resultados de la búsqueda.

Consulta PPL:

```
os> source=accounts | fields account_number, firstname, lastname;
fetched rows / total rows = 4/4
+------------------+-------------+------------+
| account_number   | firstname   | lastname   |
|------------------+-------------+------------|
| 1                | Jane        | Doe        |
| 6                | John        | Doe        |
| 13               | Jorge       | Souza      |
| 18               | Juan        | Li         |
+------------------+-------------+------------+
```

Ejemplo 2: Eliminar los campos especificados del resultado

En este ejemplo se muestra cómo quitar el campo `account_number` de los resultados de búsqueda.

Consulta PPL:

```
os> source=accounts | fields account_number, firstname, lastname | fields -
 account_number ;
fetched rows / total rows = 4/4
+-------------+------------+
| firstname   | lastname   |
|-------------+------------|
| Jane        | Doe        |
| John        | Doe        |
| Jorge       | Souza      |
| Juan        | Li         |
+-------------+------------+
```

Ejemplos adicionales

- `source = table`
- `source = table | fields a,b,c`
- `source = table | fields + a,b,c`
- `source = table | fields - b,c`
- `source = table | eval b1 = b | fields - b1,c`

Ejemplo de campos anidados:

```
`source = catalog.schema.table1, catalog.schema.table2 | fields A.nested1, B.nested1`
`source = catalog.table | where struct_col2.field1.subfield > 'valueA' | sort int_col |
 fields  int_col, struct_col.field1.subfield, struct_col2.field1.subfield`
`source = catalog.schema.table | where struct_col2.field1.subfield > 'valueA' | sort
 int_col | fields  int_col, struct_col.field1.subfield, struct_col2.field1.subfield`
```

Comando flatten

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando PPL, consulte the section called “Comandos”.

Utilice el comando flatten para expandir los campos de los siguientes tipos:

- `struct<?,?>`
- `array<struct<?,?>>`

Sintaxis

Utilice la siguiente sintaxis:

```
flatten <field>
```

- campo: el campo que se va a aplanar. El campo debe ser de un tipo compatible.

Esquema

col_name	data_type
_time	cadena
puentes	array<struct<length:bigint,name:string>>
ciudad	cadena
coor	struct<alt:bigint,lat:double,long:double>
país	cadena

Datos

_time	puentes	ciudad	coor	país
2024-09-13T12:00:00	[{801, Puente de la Torre}, {928, Puente de Londres}]	Londres	{35, 51.5074, -0.1278}	Inglaterra
2024-09-13T12:00:00	[{232, Puente Nuevo}, {160, Puente Alejandro III}]	París	{35, 48.8566, 2.3522}	Francia
2024-09-13T12:00:00	[{48, Puente de Rialto}, {11, Puente	Venecia	{2, 45.4408, 12.3155}	Italia

_time	puentes	ciudad	coor	país
	de los Suspiros}]			
2024-09-13T12:00:00	[{***, Puente de Carlos}, {343, Puente de la Legión}]	Praga	{200, 50.0755, 14.4378}	República Checa
2024-09-13T12:00:00	[{375, Puente de las Cadenas}, {333, Puente de la Libertad}]	Budapest	{96, 47.4979, 19.0402}	Hungría
1990-09-13T12:00:00	NULL	Varsovia	NULL	Polonia

Ejemplo 1: estructura aplanada

En este ejemplo se muestra cómo aplanar un campo de estructura.

Consulta PPL:

```
source=table | flatten coor
```

_time	puentes	ciudad	país	alt	lat	long
2024-09-13T12:00:00	[{801, Puente de la Torre}, {928,	Londres	Inglaterra	35	51.5074	-0.1278

_time	puentes	ciudad	país	alt	lat	long
	Puente de Londres}]					
2024-09-13T12:00:00	[{232, Puente Nuevo}, {160, Puente Alejandro III}]	París	Francia	35	48.8566	2.3522
2024-09-13T12:00:00	[{48, Puente de Rialto}, {11, Puente de los Suspiros}]	Venecia	Italia	2	45.4408	12.3155
2024-09-13T12:00:00	[{516, Puente de Carlos}, {343, Puente de la Legión}]	Praga	República Checa	200	50.0755	14.4378

_time	puentes	ciudad	país	alt	lat	long
2024-09-13T12:00:00	[{375, Puente de las Cadenas} {333, Puente de la Libertad}]	Budapest	Hungría	96	47.4979	19.0402
1990-09-13T12:00:00	NULL	Varsovia	Polonia	NULL	NULL	NULL

Ejemplo 2: aplanar una matriz

En el ejemplo se muestra cómo aplanar una matriz de campos de estructura.

Consulta PPL:

```
source=table | flatten bridges
```

_time	ciudad	coor	país	length	nombre
2024-09-13T12:00:00	Londres	{35, 51.5074, -0.1278}	Inglaterra	801	Puente de la Torre
2024-09-13T12:00:00	Londres	{35, 51.5074, -0.1278}	Inglaterra	928	Puente de Londres
2024-09-13T12:00:00	París	{35, 48.8566, 2.3522}	Francia	232	Puente Nuevo
2024-09-13T12:00:00	París	{35, 48.8566, 2.3522}	Francia	160	Puente Alejandro III

_time	ciudad	coor	país	length	nombre
2024-09-13T12:00:00	Venecia	{2, 45.4408, 12.3155}	Italia	48	Puente de Rialto
2024-09-13T12:00:00	Venecia	{2, 45.4408, 12.3155}	Italia	11	Puente de los Suspiros
2024-09-13T12:00:00	Praga	{200, 50.0755, 14.4378}	República Checa	516	Puente de Carlos
2024-09-13T12:00:00	Praga	{200, 50.0755, 14.4378}	República Checa	343	Puente de la Legión
2024-09-13T12:00:00	Budapest	{96, 47.4979, 19.0402}	Hungría	375	Puente de las Cadenas
2024-09-13T12:00:00	Budapest	{96, 47.4979, 19.0402}	Hungría	333	Puente de la Libertad
1990-09-13T12:00:00	Varsovia	NULL	Polonia	NULL	NULL

Ejemplo 3: aplanar una matriz y una estructura

En este ejemplo, se muestra cómo aplanar varios campos.

Consulta PPL:

```
source=table | flatten bridges | flatten coor
```

_time	ciudad	país	length	nombre	alt	lat	long
2024-09-13T12:00:00	Londres	Inglaterra	801	Puente de la Torre	35	51.5074	-0.1278

_time	ciudad	país	length	nombre	alt	lat	long
2024-09-13T12:00:00	Londres	Inglaterra	928	Puente de Londres	35	51.5074	-0.1278
2024-09-13T12:00:00	París	Francia	232	Puente Nuevo	35	48.8566	2.3522
2024-09-13T12:00:00	París	Francia	160	Puente Alejandro III	35	48.8566	2.3522
2024-09-13T12:00:00	Venecia	Italia	48	Puente de Rialto	2	45.4408	12.3155
2024-09-13T12:00:00	Venecia	Italia	11	Puente de los Suspiros	2	45.4408	12.3155
2024-09-13T12:00:00	Praga	República Checa	516	Puente de Carlos	200	50.0755	14.4378
2024-09-13T12:00:00	Praga	República Checa	343	Puente de la Legión	200	50.0755	14.4378
2024-09-13T12:00:00	Budape	Hungría	375	Puente de las Cadenas	96	47.4979	19.0402
2024-09-13T12:00:00	Budape	Hungría	333	Puente de la Libertad	96	47.4979	19.0402

_time	ciudad	país	length	nombre	alt	lat	long
1990-09-13T12:00:00	Varsovi	Polonia	NULL	NULL	NULL	NULL	NULL

Comando grok

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando PPL, consulte the section called “Comandos”.

El comando `grok` analiza un campo de texto con un patrón grok y agrega los resultados al resultado de la búsqueda.

Sintaxis

Utilice la siguiente sintaxis:

```
grok <field> <pattern>
```

campo

- Obligatorio.
- El campo debe ser un campo de texto.

pattern

- Obligatorio.
- El patrón grok utilizado para extraer nuevos campos del campo de texto dado.
- Si ya existe un nombre de campo nuevo, reemplazará al campo original.

Patrón de Grok

El patrón grok se utiliza para hacer coincidir el campo de texto de cada documento y extraer nuevos campos.

Ejemplo 1: creación de un campo nuevo

En este ejemplo se muestra cómo crear un nuevo campo `host` para cada documento. `host` será el nombre de host que aparece después `@` en el campo `email`. Si se analiza un campo nulo, se devuelve una cadena vacía.

```
os> source=accounts | grok email '.+@%{HOSTNAME:host}' | fields email, host ;
fetched rows / total rows = 4/4
+-------------------------+-------------+
| email                   | host        |
|-------------------------+-------------|
| jane_doe@example.com    | example.com |
| arnav_desai@example.net | example.net |
| null                    |             |
| juan_li@example.org     | example.org |
+-------------------------+-------------+
```

Ejemplo 2: anular el campo existente

En este ejemplo se muestra cómo anular el campo `address` existente quitando el número de calle.

```
os> source=accounts | grok address '%{NUMBER} %{GREEDYDATA:address}' | fields address ;
fetched rows / total rows = 4/4
+------------------+
| address          |
|------------------|
| Example Lane     |
| Any Street       |
| Main Street      |
| Example Court    |
+------------------+
```

Ejemplo 3: Uso de grok para analizar registros

En este ejemplo, se muestra cómo utilizar grok para analizar registros sin procesar.

```
os> source=apache | grok message '%{COMMONAPACHELOG}' | fields COMMONAPACHELOG,
 timestamp, response, bytes ;
fetched rows / total rows = 4/4
```

```
+-----------------------------------------------------------------------------------------------------------
+---------------------------+-----------+---------+
| COMMONAPACHELOG
                                          | timestamp                  | response   |
 bytes   |
|-----------------------------------------------------------------------------------------------------------
+---------------------------+-----------+---------|
| 177.95.8.74 - upton5450 [28/Sep/2022:10:15:57 -0700] "HEAD /e-business/mindshare
 HTTP/1.0" 404 19927                       | 28/Sep/2022:10:15:57 -0700 | 404        |
 19927   |
| 127.45.152.6 - pouros8756 [28/Sep/2022:10:15:57 -0700] "GET /architectures/
convergence/niches/mindshare HTTP/1.0" 100 28722 | 28/Sep/2022:10:15:57 -0700 | 100
    | 28722   |
| *************** - - [28/Sep/2022:10:15:57 -0700] "PATCH /strategize/out-of-the-box
 HTTP/1.0" 401 27439                      | 28/Sep/2022:10:15:57 -0700 | 401        |
 27439   |
| ************** - - [28/Sep/2022:10:15:57 -0700] "POST /users HTTP/1.1" 301 9481
                                        | 28/Sep/2022:10:15:57 -0700 | 301        | 9481
    |
+-----------------------------------------------------------------------------------------------------------
+---------------------------+-----------+---------+
```

Limitaciones

El comando grok tiene las mismas limitaciones que el comando parse.

Comando head

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando PPL, consulte the section called “Comandos”.

Utilice el comando `head` para devolver el primer número N de resultados especificados después de un desplazamiento opcional en el orden de búsqueda.

Sintaxis

Utilice la siguiente sintaxis:

```
head [<size>] [from <offset>]
```

<size>

- Número entero opcional.
- El número de resultados que se devolverá.
- Valor predeterminado: 10

<offset>

- Entero después de opcional `from`.
- Número de resultados que se va a saltear.
- Predeterminado: 0

Ejemplo 1: Obtenga los primeros 10 resultados

En este ejemplo se muestra cómo recuperar un máximo de 10 resultados del índice de cuentas.

Consulta PPL:

```
os> source=accounts | fields firstname, age | head;
fetched rows / total rows = 4/4
+-------------+-------+
| firstname   | age   |
|-------------+-------|
| Jane        | 32    |
| John        | 36    |
| Jorge       | 28    |
| Juan        | 33    |
+-------------+-------+
```

Ejemplo 2: Obtenga los primeros N resultados

El ejemplo muestra los primeros N resultados del índice de cuentas.

Consulta PPL:

```
os> source=accounts | fields firstname, age | head 3;
fetched rows / total rows = 3/3
+-------------+-------+
| firstname   | age   |
```

```
|-------------+-------|
| Jane        | 32    |
| John        | 36    |
| Jorge       | 28    |
+-------------+-------+
```

Ejemplo 3: Obtenga los primeros N resultados después de la compensación M

En este ejemplo se muestra cómo recuperar los primeros N resultados después de omitir M resultados del índice de cuentas.

Consulta PPL:

```
os> source=accounts | fields firstname, age | head 3 from 1;
fetched rows / total rows = 3/3
+-------------+-------+
| firstname   | age   |
|-------------+-------|
| John        | 36    |
| Jorge       | 28    |
| Juan        | 33    |
+-------------+-------+
```

Comando join

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando PPL, consulte the section called “Comandos”.

El comando join le permite combinar datos de varios orígenes en función de campos comunes, lo que le permite realizar análisis complejos y obtener información más detallada a partir de sus conjuntos de datos distribuidos.

Esquema

Hay al menos dos índices: `otel-v1-apm-span-*` (grande) y `otel-v1-apm-service-map` (pequeño).

Campos relevantes de los índices:

otel-v1-apm-span-*

- traceId: un identificador único para un rastreo. Todos los tramos de la misma traza comparten el mismo traceId.
- spanId: identificador único para un tramo dentro de una traza, que se asigna cuando se crea el tramo.
- parentSpanId: el spanId del grupo principal de este tramo. Si se trata de un intervalo raíz, este campo debe estar vacío.
- durationInNanos: la diferencia en nanosegundos entre startTime y endTime. (esto está en la interfaz de usuario `latency`)
- serviceName: el recurso desde el que se origina el tramo.
- traceGroup: nombre del tramo raíz de la traza.

otel-v1-apm-service-map

- serviceName: nombre del servicio que registró el intervalo.
- destination.domain: el serviceName del servicio que llama este cliente.
- destination.resource: el nombre del intervalo (API, operación, etc.) al que llama este cliente.
- target.domain: el serviceName del servicio que llama un cliente.
- target.resource: el nombre del intervalo (API, operación, etc.) al que llama un cliente.
- traceGroupName: el nombre del intervalo de nivel superior que inició la cadena de solicitudes.

Requisito

Admita join para calcular lo siguiente:

Para cada servicio, combine el índice span con el índice del mapa de servicios para calcular las métricas con distintos tipos de filtros.

Esta consulta de ejemplo calcula la latencia al filtrarla por grupo de rastreo `client_cancel_order` para el servicio `order`.

```
SELECT avg(durationInNanos)
FROM `otel-v1-apm-span-000001` t1
WHERE t1.serviceName = `order`
  AND ((t1.name in
          (SELECT target.resource
```

```
           FROM `otel-v1-apm-service-map`
           WHERE serviceName = `order`
             AND traceGroupName = `client_cancel_order`)
       AND t1.parentSpanId != NULL)
      OR (t1.parentSpanId = NULL
          AND t1.name = `client_cancel_order`))
  AND t1.traceId in
    (SELECT traceId
     FROM `otel-v1-apm-span-000001`
     WHERE serviceName = `order`)
```

Migración a PPL

Sintaxis del comando join

```
SEARCH source=<left-table>
| <other piped command>
| [joinType] JOIN
    [leftAlias]
    ON joinCriteria
    <right-table>
| <other piped command>
```

Reescrituras

```
SEARCH source=otel-v1-apm-span-000001
| WHERE serviceName = 'order'
| JOIN left=t1 right=t2
    ON t1.traceId = t2.traceId AND t2.serviceName = 'order'
    otel-v1-apm-span-000001 -- self inner join
| EVAL s_name = t1.name -- rename to avoid ambiguous
| EVAL s_parentSpanId = t1.parentSpanId -- RENAME command would be better when it is
 supported
| EVAL s_durationInNanos = t1.durationInNanos
| FIELDS s_name, s_parentSpanId, s_durationInNanos -- reduce colunms in join
| LEFT JOIN left=s1 right=t3
    ON s_name = t3.target.resource AND t3.serviceName = 'order' AND t3.traceGroupName =
 'client_cancel_order'
    otel-v1-apm-service-map
| WHERE (s_parentSpanId IS NOT NULL OR (s_parentSpanId IS NULL AND s_name =
 'client_cancel_order'))
| STATS avg(s_durationInNanos) -- no need to add alias if there is no ambiguous
```

joinType

- Sintaxis: `INNER | LEFT OUTER | CROSS`
- Opcional
- El tipo de unión que se va a realizar. Si no se especifica, el valor predeterminado es `INNER`.

leftAlias

- Sintaxis: `left = <leftAlias>`
- Opcional
- El alias de la subconsulta que se debe usar con el lado izquierdo de la combinación, para evitar nombres ambiguos.

joinCriteria

- Sintaxis: `<expression>`
- Obligatorio
- La sintaxis comienza con `ON`. Puede ser cualquier expresión de comparación. En general, los criterios de unión son los siguientes `<leftAlias>.<leftField>=<rightAlias>.<rightField>`.

  Por ejemplo: `l.id = r.id`. Si los criterios de unión contienen varias condiciones, puede especificar operador `AND` y `OR` entre cada expresión de comparación. Por ejemplo, `l.id = r.id AND l.email = r.email AND (r.age > 65 OR r.age < 18)`.

Más ejemplos

Migración desde una consulta SQL (TPC-H Q13):

```
SELECT c_count, COUNT(*) AS custdist
FROM
  ( SELECT c_custkey, COUNT(o_orderkey) c_count
    FROM customer LEFT OUTER JOIN orders ON c_custkey = o_custkey
        AND o_comment NOT LIKE '%unusual%packages%'
    GROUP BY c_custkey
  ) AS c_orders
GROUP BY c_count
```

```
ORDER BY custdist DESC, c_count DESC;
```

Reescrito por PPL Join Query:

```
SEARCH source=customer
| FIELDS c_custkey
| LEFT OUTER JOIN
    ON c_custkey = o_custkey AND o_comment NOT LIKE '%unusual%packages%'
    orders
| STATS count(o_orderkey) AS c_count BY c_custkey
| STATS count() AS custdist BY c_count
| SORT - custdist, - c_count
```

Limitación: no se admiten subbúsquedas en el lado derecho de la unión.

Si se admiten las subbúsquedas, puedes reescribir la consulta PPL anterior de la siguiente manera:

```
SEARCH source=customer
| FIELDS c_custkey
| LEFT OUTER JOIN
   ON c_custkey = o_custkey
   [
      SEARCH source=orders
      | WHERE o_comment NOT LIKE '%unusual%packages%'
      | FIELDS o_orderkey, o_custkey
   ]
| STATS count(o_orderkey) AS c_count BY c_custkey
| STATS count() AS custdist BY c_count
| SORT - custdist, - c_count
```

Comando lookup

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando PPL, consulte the section called “Comandos”.

Utilice el comando `lookup` para enriquecer los datos de búsqueda mediante la adición o sustitución de datos de un índice de búsqueda (tabla de dimensiones). Este comando le permite extender los

campos de un índice con valores de una tabla de dimensiones. También se puede utilizar para agregar o reemplazar valores cuando se cumplen las condiciones de búsqueda. El comando `lookup` es más adecuado que el comando `Join` para enriquecer los datos de origen con un conjunto de datos estático.

Sintaxis

Utilice la siguiente sintaxis:

```
SEARCH source=<sourceIndex>
| <other piped command>
| LOOKUP <lookupIndex> (<lookupMappingField> [AS <sourceMappingField>])...
    [(REPLACE | APPEND) (<inputField> [AS <outputField>])...]
| <other piped command>
```

lookupIndex

- Obligatorio.
- El nombre del índice de búsqueda (tabla de dimensiones).

lookupMappingField

- Obligatorio.
- Una clave de asignación en el índice de búsqueda, análoga a una clave de unión de la tabla de la derecha. Puede especificar múltiples nombres de esquema separados por comas.

sourceMappingField

- Opcional.
- Predeterminado: <lookupMappingField>.
- Una clave de asignación de la consulta de origen, análoga a una clave de unión del lado izquierdo.

inputField

- Opcional.
- Predeterminado: todos los campos del índice de búsqueda en los que se encuentran valores coincidentes.

- Campo del índice de búsqueda en el que los valores coincidentes se aplican a la salida del resultado. Puede especificar múltiples nombres de esquema separados por comas.

outputField

- Opcional.
- Valor predeterminado: `<inputField>`.
- Un campo de salida. Puede especificar varios campos de salida. Si especifica un nombre de campo existente en la consulta de origen, sus valores se sustituirán o se agregarán por valores coincidentes de inputField. Si especifica un nombre de campo nuevo, se agregará a los resultados.

REPLACE | APPEND

- Opcional.
- Predeterminado: REPLACE
- Especifica cómo administrar los valores coincidentes. Si especifica REPLACE, los valores coincidentes del campo **<lookupIndex>** sobrescribirán los valores del resultado. Si especifica `APPEND`, los valores coincidentes del campo **<lookupIndex>** solo se incorporarán a los valores faltantes en el resultado.

Uso

- LOOKUP **<lookupIndex>** id AS cid REPLACE mail AS email
- LOOKUP **<lookupIndex>** name REPLACE mail AS email
- LOOKUP **<lookupIndex>** id AS cid, name APPEND address, mail AS email
- LOOKUP **<lookupIndex>** id

Ejemplo

Vea los siguientes ejemplos.

```
SEARCH source=<sourceIndex>
| WHERE orderType = 'Cancelled'
| LOOKUP account_list, mkt_id AS mkt_code REPLACE amount, account_name AS name
| STATS count(mkt_code), avg(amount) BY name
```

```
SEARCH source=<sourceIndex>
| DEDUP market_id
| EVAL category=replace(category, "-", ".")
| EVAL category=ltrim(category, "dvp.")
| LOOKUP bounce_category category AS category APPEND classification
```

```
SEARCH source=<sourceIndex>
| LOOKUP bounce_category category
```

Comando parse

El comando `parse` analiza un campo de texto con una expresión regular y agrega el resultado al resultado de la búsqueda.

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando PPL, consulte the section called “Comandos”.

Sintaxis

Utilice la siguiente sintaxis:

```
parse <field> <pattern>
```

`field`

- Obligatorio.
- El campo debe ser un campo de texto.

`pattern`

- Cadena obligatoria.
- Este es el patrón de expresión regular que se utiliza para extraer nuevos campos del campo de texto dado.
- Si ya existe un nombre de campo nuevo, reemplazará al campo original.

Expresión regular

El patrón de expresiones regulares se utiliza para hacer coincidir todo el campo de texto de cada documento con el motor de expresiones regulares de Java. Cada grupo de captura nombrado en la expresión se convertirá en un campo nuevo `STRING`.

Ejemplo 1: creación de un campo nuevo

En el ejemplo se muestra cómo crear un nuevo campo `host` para cada documento. `host` será el nombre del host que aparece después `@` en el campo `email`. Si se analiza un campo nulo, se devuelve una cadena vacía.

Consulta PPL:

```
os> source=accounts | parse email '.+@(?<host>.+)' | fields email, host ;
fetched rows / total rows = 4/4
+-----------------------+-------------+
| email                 | host        |
|-----------------------+-------------|
| jane_doe@example.com  | example.com |
| john_doe@example.net  | example.net |
| null                  |             |
| juan_li@example.org   | example.org |
+-----------------------+-------------+
```

Ejemplo 2: Anular un campo existente

El ejemplo muestra cómo anular el campo `address` existente quitando el número de calle.

Consulta PPL:

```
os> source=accounts | parse address '\d+ (?<address>.+)' | fields address ;
fetched rows / total rows = 4/4
+------------------+
| address          |
|------------------|
| Example Lane     |
| Example Street   |
| Example Avenue   |
| Example Court    |
+------------------+
```

Ejemplo 3: Filtrar y ordenar por campo analizado

En el ejemplo se muestra cómo ordenar los números de calle superiores a 500 en el campo `address`.

Consulta PPL:

```
os> source=accounts | parse address '(?<streetNumber>\d+) (?<street>.+)' | where
 cast(streetNumber as int) > 500 | sort num(streetNumber) | fields streetNumber,
 street ;
fetched rows / total rows = 3/3
+----------------+----------------+
| streetNumber   | street         |
|----------------+----------------|
| ***            | Example Street |
| ***            | Example Avenue |
| 880            | Example Lane   |
+----------------+----------------+
```

Limitaciones

El comando parse presenta algunas limitaciones:

- Los campos definidos por parse no se pueden volver a analizar.

  Este comando no funcionara:

  ```
  source=accounts | parse address '\d+ (?<street>.+)' | parse street '\w+ (?<road>\w+)'
  ```

- Los campos definidos por parse no se pueden anular con otros comandos.

  `where` no coincidirá con ningún documento ya que `street` no se puede anular:

  ```
  source=accounts | parse address '\d+ (?<street>.+)' | eval street='1' | where
   street='1' ;
  ```

- El campo de texto utilizado por parse no se puede anular.

  `street` no se analizará correctamente ya que `address` está anulado:

  ```
  source=accounts | parse address '\d+ (?<street>.+)' | eval address='1' ;
  ```

- Los campos definidos por parse no se pueden filtrar ni ordenar después de usarlos en el comando `stats`.

`where` en el siguiente comando no funcionará:

```
source=accounts | parse email '.+@(?<host>.+)' | stats avg(age) by host | where
 host=pyrami.com ;
```

Comando patterns

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando PPL, consulte the section called “Comandos”.

El comando `patterns` extrae los patrones de registro de un campo de texto y agrega los resultados al resultado de la búsqueda. Al agrupar los registros por sus patrones, resulta más fácil agregar estadísticas de grandes volúmenes de datos de registro para analizarlos y solucionar problemas.

Sintaxis

Utilice la siguiente sintaxis:

```
patterns [new_field=<new-field-name>] [pattern=<pattern>] <field>
```

new-field-name

- Cadena opcional.
- Este es el nombre del nuevo campo para los patrones extraídos.
- El valor predeterminado es `patterns_field`.
- Si el nombre ya existe, sustituirá al campo original.

pattern

- Cadena opcional.
- Este es el patrón de caracteres regex que deben filtrarse del campo de texto.
- Si no está presente, el patrón predeterminado son los caracteres alfanuméricos (`[a-zA-Z\d]`).

campo

- Obligatorio.
- El campo debe ser un campo de texto.

Ejemplo 1: creación de un campo nuevo

En el ejemplo se muestra cómo utilizar los signos de puntuación extraídos en `email` cada documento. Si se analiza un campo nulo, se devuelve una cadena vacía.

Consulta PPL:

```
os> source=accounts | patterns email | fields email, patterns_field ;
fetched rows / total rows = 4/4
+-----------------------+------------------+
| email                 | patterns_field   |
|-----------------------+------------------|
| jane_doe@example.com  | @.               |
| john_doe@example.net  | @.               |
| null                  |                  |
| juan_li@example.org   | @.               |
+-----------------------+------------------+
```

Ejemplo 2: Extraer patrones de registro

El ejemplo muestra cómo extraer signos de puntuación de un campo de registro sin procesar utilizando los patrones predeterminados.

Consulta PPL:

```
os> source=apache | patterns message | fields message, patterns_field ;
fetched rows / total rows = 4/4
+-------------------------------------------------------------------------------------------------
+---------------------------------+
| message
                                       | patterns_field                |
|-------------------------------------------------------------------------------------------------
+---------------------------------|
| 177.95.8.74 - upton5450 [28/Sep/2022:10:15:57 -0700] "HEAD /e-business/mindshare
 HTTP/1.0" 404 19927                          | ... -  [//::: -] " /-/ /."      |
| ************ - pouros8756 [28/Sep/2022:10:15:57 -0700] "GET /architectures/
convergence/niches/mindshare HTTP/1.0" 100 28722 | ... -  [//::: -] " //// /."     |
```

```
| *************** - - [28/Sep/2022:10:15:57 -0700] "PATCH /strategize/out-of-the-box
 HTTP/1.0" 401 27439                      | ... - - [//::: -] " //--- /."   |
| ************** - - [28/Sep/2022:10:15:57 -0700] "POST /users HTTP/1.1" 301 9481
                                        | ... - - [//::: -] " / /."       |
+-----------------------------------------------------------------------------------------
+--------------------------------+
```

Ejemplo 3: Extraiga patrones de registro con un patrón de expresiones regulares personalizado

El ejemplo muestra cómo extraer signos de puntuación de un campo de registro sin procesar utilizando patrones definidos por el usuario.

Consulta PPL:

```
os> source=apache | patterns new_field='no_numbers' pattern='[0-9]' message | fields
 message, no_numbers ;
fetched rows / total rows = 4/4
+-----------------------------------------------------------------------------------------
+-------------------------------------------------------------------------------
+
| message
                                        | no_numbers
                                        |
|-----------------------------------------------------------------------------------------
+-------------------------------------------------------------------------------|
| 177.95.8.74 - upton5450 [28/Sep/2022:10:15:57 -0700] "HEAD /e-business/mindshare
 HTTP/1.0" 404 19927                         | ... - upton [/Sep/::: -] "HEAD /e-
business/mindshare HTTP/."                           |
| 127.45.152.6 - pouros8756 [28/Sep/2022:10:15:57 -0700] "GET /architectures/
convergence/niches/mindshare HTTP/1.0" 100 28722 | ... - pouros [/Sep/::: -] "GET /
architectures/convergence/niches/mindshare HTTP/."   |
| *************** - - [28/Sep/2022:10:15:57 -0700] "PATCH /strategize/out-of-the-box
 HTTP/1.0" 401 27439                      | ... - - [/Sep/::: -] "PATCH /strategize/
out-of-the-box HTTP/."                         |
| ************** - - [28/Sep/2022:10:15:57 -0700] "POST /users HTTP/1.1" 301 9481
                                        | ... - - [/Sep/::: -] "POST /users HTTP/."
                                        |
+-----------------------------------------------------------------------------------------
+-------------------------------------------------------------------------------
+
```

Limitación

El comando patterns tiene las mismas limitaciones que el comando parse.

Comando rare

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando PPL, consulte the section called “Comandos”.

Utilice el comando `rare` para buscar la tupla de valores menos común de todos los campos de la lista de campos.

> **Note**
>
> Se devuelve un máximo de 10 resultados por cada tupla distinta de valores de los campos agrupados.

Sintaxis

Utilice la siguiente sintaxis:

```
rare [N] <field-list> [by-clause] rare_approx [N] <field-list> [by-clause]
```

field-list

- Obligatorio.
- Una lista delimitada por comas de nombres de campo.

by-clause

- Opcional.
- Uno o más campos por los que agrupar los resultados.

N

- El número de resultados que se devolverá.

- Valor predeterminado: 10

rare_approx

- El recuento aproximado de los (n) campos raros mediante el uso de la cardinalidad estimada mediante el algoritmo HyperLogLog++.

Ejemplo 1: Encuentre los valores menos comunes en un campo

El ejemplo busca el género menos común de todas las cuentas.

Consulta PPL:

```
os> source=accounts | rare gender;
os> source=accounts | rare_approx 10 gender;
os> source=accounts | rare_approx gender;
fetched rows / total rows = 2/2
+----------+
| gender   |
|----------|
| F        |
| M        |
+----------+
```

Ejemplo 2: encuentra los valores menos comunes organizados por género

En el ejemplo, se busca la edad menos común de todas las cuentas agrupadas por género.

Consulta PPL:

```
os> source=accounts | rare 5 age by gender;
os> source=accounts | rare_approx 5 age by gender;
fetched rows / total rows = 4/4
+----------+-------+
| gender   | age   |
|----------+-------|
| F        | 28    |
| M        | 32    |
| M        | 33    |
| M        | 36    |
+----------+-------+
```

Comando rename

Utilice el comando `rename` para cambiar los nombres de uno o más campos del resultado de la búsqueda.

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando PPL, consulte the section called “Comandos”.

Sintaxis

Utilice la siguiente sintaxis:

```
rename <source-field> AS <target-field>["," <source-field> AS <target-field>]...
```

source-field

- Obligatorio.
- Este es el nombre del campo que desea cambiar de nombre.

target-field

- Obligatorio.
- Este es el nombre al que quiere cambiarle el nombre.

Ejemplo 1: cambiar el nombre de un campo

En este ejemplo, se muestra cómo renombrar un único campo.

Consulta PPL:

```
os> source=accounts | rename account_number as an | fields an;
fetched rows / total rows = 4/4
+------+
| an   |
|------|
| 1    |
| 6    |
```

```
| 13   |
| 18   |
+------+
```

Ejemplo 2: cambiar el nombre de varios campos

En este ejemplo, se muestra cómo renombrar varios campos.

Consulta PPL:

```
os> source=accounts | rename account_number as an, employer as emp | fields an, emp;
fetched rows / total rows = 4/4
+------+---------+
| an   | emp     |
|------+---------|
| 1    | Pyrami  |
| 6    | Netagy  |
| 13   | Quility |
| 18   | null    |
+------+---------+
```

Limitaciones

- No se admite la sustitución de un campo existente:

```
source=accounts | grok address '%{NUMBER} %{GREEDYDATA:address}' | fields address
```

Comando search

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando PPL, consulte the section called “Comandos”.

Utilice el comando `search` para recuperar documentos de un índice. El comando `search` solo se puede usar como primer comando en una consulta PPL.

Sintaxis

Utilice la siguiente sintaxis:

```
search source=[<remote-cluster>:]<index> [boolean-expression]
```

buscar

- Opcional.
- Busca palabras clave, que se pueden omitir.

índice

- Obligatorio.
- El comando de búsqueda debe especificar el índice desde el que realizar la consulta.
- El nombre del índice puede tener un prefijo `<cluster name>:` para las búsquedas entre clústeres.

bool-expression

- Opcional.
- Una cadena de expresión que se evalúa como un valor booleano.

Ejemplo 1: obtener todos los datos

El ejemplo muestra cómo recuperar todo el documento del índice de cuentas.

Consulta PPL:

```
os> source=accounts;
+------------------+-------------+----------------------+-----------+----------
+--------+----------------+---------+-------+-----------------------+------------+
| account_number   | firstname   | address              | balance   | gender   | city
 | employer       | state   | age   | email                 | lastname   |
|------------------+-------------+----------------------+-----------+----------
+--------+----------------+---------+-------+-----------------------+------------|
| 1                | Jorge       | *** Any Lane         | 39225     | M        | Brogan
 | ExampleCorp    | IL      | 32    | jane_doe@example.com  | Souza      |
| 6                | John        | *** Example Street   | 5686      | M        | Dante
 | AnyCorp        | TN      | 36    | john_doe@example.com  | Doe        |
```

```
| 13               | Jane        | *** Any Street       | *****     | F        | Nogal
 | ExampleCompany | VA      | 28    | null                  | Doe        |
| 18               | Juan        | *** Example Court    | 4180      | M        | Orick
 | null           | MD      | 33    | juan_li@example.org   | Li         |
+------------------+-------------+----------------------+-----------+----------
+--------+----------------+---------+-------+----------------------+------------+
```

Ejemplo 2: Obtener datos con una condición

El ejemplo muestra cómo recuperar todo el documento del índice de cuentas con .

Consulta PPL:

```
os> SEARCH source=accounts account_number=1 or gender="F";
+------------------+-------------+--------------------+-----------+----------+--------
+----------------+---------+-------+-------------------------+------------+
| account_number   | firstname   | address            | balance   | gender   | city   |
 employer       | state   | age   | email                 -  | lastname   |
|------------------+-------------+--------------------+-----------+----------+--------
+----------------+---------+-------+-------------------------+------------|
| 1                | Jorge       | *** Any Lane       | *****     | M        | Brogan |
 ExampleCorp    | IL      | 32    | jorge_souza@example.com | Souza      |
| 13               | Jane        | *** Any Street     | *****     | F        | Nogal  |
 ExampleCompany | VA      | 28    | null                    | Doe        |
+------------------+-------------+--------------------+-----------+----------+--------
+-----------------+---------+-------+-------------------------+------------+
```

Comando sort

Utilice el comando `sort` para ordenar los resultados de la búsqueda por campos específicos.

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando PPL, consulte the section called “Comandos”.

Sintaxis

Utilice la siguiente sintaxis:

```
sort <[+|-] sort-field>...
```

[+|-]

- Opcional.
- El signo más [+] indica el orden ascendente con los valores NULL o MISSING al principio.
- El signo menos [-] indica el orden descendente con los valores NULL o MISSING al final.
- Predeterminado: orden ascendente con los valores NULL o MISSING primero.

sort-field

- Obligatorio.
- El campo que se utiliza para ordenar.

Ejemplo 1: Ordenar por un campo

El ejemplo muestra cómo ordenar el documento con el campo de edad en orden ascendente.

Consulta PPL:

```
os> source=accounts | sort age | fields account_number, age;
fetched rows / total rows = 4/4
+------------------+-------+
| account_number   | age   |
|------------------+-------|
| 13               | 28    |
| 1                | 32    |
| 18               | 33    |
| 6                | 36    |
+------------------+-------+
```

Ejemplo 2: ordenar por un campo y devolver todos los resultados

El ejemplo muestra cómo ordenar el documento con el campo de edad en orden ascendente.

Consulta PPL:

```
os> source=accounts | sort age | fields account_number, age;
fetched rows / total rows = 4/4
```

```
+------------------+-------+
| account_number   | age   |
|------------------+-------|
| 13               | 28    |
| 1                | 32    |
| 18               | 33    |
| 6                | 36    |
+------------------+-------+
```

Ejemplo 3: ordenar por un campo en orden descendente

El ejemplo muestra cómo ordenar el documento con el campo de edad en orden descendente.

Consulta PPL:

```
os> source=accounts | sort - age | fields account_number, age;
fetched rows / total rows = 4/4
+------------------+-------+
| account_number   | age   |
|------------------+-------|
| 6                | 36    |
| 18               | 33    |
| 1                | 32    |
| 13               | 28    |
+------------------+-------+
```

Ejemplo 4: Ordenar por varios campos

El ejemplo muestra cómo ordenar el documento con el campo de género en orden ascendente y el campo de edad en orden descendente.

Consulta PPL:

```
os> source=accounts | sort + gender, - age | fields account_number, gender, age;
fetched rows / total rows = 4/4
+------------------+----------+-------+
| account_number   | gender   | age   |
|------------------+----------+-------|
| 13               | F        | 28    |
| 6                | M        | 36    |
| 18               | M        | 33    |
| 1                | M        | 32    |
+------------------+----------+-------+
```

Ejemplo 5: Ordenar por campo e incluir un valor nulo

En el ejemplo se muestra cómo ordenar el campo del empleador por la opción predeterminada (en orden ascendente y con los valores nulo primero). El resultado muestra que el valor nulo está en la primera fila.

Consulta PPL:

```
os> source=accounts | sort employer | fields employer;
fetched rows / total rows = 4/4
+------------+
| employer   |
|------------|
| null       |
| AnyCompany |
| AnyCorp    |
| AnyOrgty   |
+------------+
```

Comandos stats

Utilice el comando `stats` para calcular la agregación a partir del resultado de la búsqueda.

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando PPL, consulte the section called “Comandos”.

Manejo de valores NULL o MISSING

Manejo de valores NULL o MISSING

Función	NULL	AUSENTE
COUNT	No contado	No contado
SUM	Ignore	Ignore
AVG	Ignore	Ignore

Función	NULL	AUSENTE
MAX	Ignore	Ignore
MIN	Ignore	Ignore

Sintaxis

Utilice la siguiente sintaxis:

```
stats <aggregation>... [by-clause]
```

agregación

- Obligatorio.
- Función de agregación aplicada a un campo.

by-clause

- Opcional.
- Sintaxis: `by [span-expression,] [field,]...`
- Especifica los campos y expresiones para agrupar los resultados de la agregación. La cláusula secundaria le permite agrupar los resultados de agregación mediante campos y expresiones. Puede utilizar funciones escalares, funciones de agregación e incluso expresiones de extensión para dividir campos específicos en grupos de intervalos iguales.
- Predeterminado: si `<by-clause>` no se especifica, el comando stats devuelve una sola fila que representa la agregación de todo el conjunto de resultados.

span-expression

- Opcional, como máximo uno.
- Sintaxis: `span(field_expr, interval_expr)`
- La unidad de la expresión de intervalo es la unidad natural por defecto. Si el campo es de tipo fecha y hora y el intervalo está expresado en unidades de fecha y hora, especifique la unidad en la expresión del intervalo.

- Por ejemplo, parece que se divide el campo `age` en buckets de 10 años `span(age, 10)`. Para dividir un campo de fecha y hora en intervalos de una hora, utilice `span(timestamp, 1h)`.

Unidades de tiempo disponibles

Unidades de intervalo
milisegundo (ms)
segundo (s)
minuto (m, distingue entre mayúsculas y minúsculas)
hora (h)
día (d)
semana (s)
mes (M, distingue entre mayúsculas y minúsculas)
trimestre (q)
año (y)

Funciones de agregación

COUNT

Devuelve un recuento del número de expr en las filas recuperadas por una sentencia SELECT.

Ejemplo:

```
os> source=accounts | stats count();
fetched rows / total rows = 1/1
+-----------+
| count()   |
|-----------|
| 4         |
+-----------+
```

SUM

Se utiliza `SUM(expr)` para devolver la suma de expr.

Ejemplo

```
os> source=accounts | stats sum(age) by gender;
fetched rows / total rows = 2/2
+------------+----------+
| sum(age)   | gender   |
|------------+----------|
| 28         | F        |
| 101        | M        |
+------------+----------+
```

AVG

Se utiliza `AVG(expr)` para devolver el valor medio de la expr.

Ejemplo

```
os> source=accounts | stats avg(age) by gender;
fetched rows / total rows = 2/2
+--------------------+----------+
| avg(age)           | gender   |
|--------------------+----------|
| 28.0               | F        |
| 33.666666666666664 | M        |
+--------------------+----------+
```

MAX

Se utiliza `MAX(expr)` para devolver el valor máximo de expr.

Ejemplo

```
os> source=accounts | stats max(age);
fetched rows / total rows = 1/1
+------------+
| max(age)   |
|------------|
| 36         |
```

```
+------------+
```

MIN

Se utiliza `MIN(expr)` para devolver el valor mínimo de expr.

Ejemplo

```
os> source=accounts | stats min(age);
fetched rows / total rows = 1/1
+------------+
| min(age)   |
|------------|
| 28         |
+------------+
```

STDDEV_SAMP

Se utiliza `STDDEV_SAMP(expr)` para devolver la desviación estándar muestral de expr.

Ejemplo:

```
os> source=accounts | stats stddev_samp(age);
fetched rows / total rows = 1/1
+--------------------+
| stddev_samp(age)   |
|--------------------|
| 3.304037933599835  |
+--------------------+
```

STDDEV_POP

Se utiliza `STDDEV_POP(expr)` para devolver la desviación estándar poblacional de expr.

Ejemplo:

```
os> source=accounts | stats stddev_pop(age);
fetched rows / total rows = 1/1
+--------------------+
| stddev_pop(age)    |
|--------------------|
| 2.**************** |
```

```
+--------------------+
```

TAKE

Se utiliza `TAKE(field [, size])` para devolver los valores originales de un campo. No garantiza el orden de los valores.

campo

- Obligatorio.
- El campo debe ser un campo de texto.

tamaño

- Número entero opcional.
- Debe devolverse el número de valores.
- El valor predeterminado es 10.

Ejemplo

```
os> source=accounts | stats take(firstname);
fetched rows / total rows = 1/1
+-----------------------------+
| take(firstname)             |
|-----------------------------|
| [Jane, Mary, Nikki, Juan    |
+-----------------------------+
```

PERCENTILE o PERCENTILE_APPROX

Utilice `PERCENTILE(expr, percent)` o `PERCENTILE_APPROX(expr, percent)` para devolver el valor percentil aproximado de expr en el porcentaje especificado.

porcentaje

- El número debe ser una constante entre 0 y 100.

Ejemplo

```
os> source=accounts | stats percentile(age, 90) by gender;
fetched rows / total rows = 2/2
+----------------------+----------+
| percentile(age, 90)   | gender   |
|----------------------+----------|
| 28                    | F        |
| 36                    | M        |
+----------------------+----------+
```

Ejemplo 1: Calcular el recuento de eventos

El ejemplo muestra cómo calcular el recuento de eventos en las cuentas.

```
os> source=accounts | stats count();
fetched rows / total rows = 1/1
+-----------+
| count()   |
|-----------|
| 4         |
+-----------+
```

Ejemplo 2: Calcular el promedio de un campo

En el ejemplo se muestra cómo calcular la edad media de todas las cuentas.

```
os> source=accounts | stats avg(age);
fetched rows / total rows = 1/1
+------------+
| avg(age)   |
|------------|
| 32.25      |
+------------+
```

Ejemplo 3: Calcular el promedio de un campo por grupo

El ejemplo muestra cómo calcular la edad media de todas las cuentas, agrupadas por género.

```
os> source=accounts | stats avg(age) by gender;
fetched rows / total rows = 2/2
+--------------------+----------+
| avg(age)           | gender   |
|--------------------+----------|
```

```
| 28.0               | F        |
| 33.666666666666664 | M        |
+--------------------+----------+
```

Ejemplo 4: Calcular el promedio, la suma y el recuento de un campo por grupo

En el ejemplo se muestra cómo calcular la edad media, la edad sumada y el recuento de eventos de todas las cuentas, agrupadas por género.

```
os> source=accounts | stats avg(age), sum(age), count() by gender;
fetched rows / total rows = 2/2
+--------------------+------------+-----------+----------+
| avg(age)           | sum(age)   | count()   | gender   |
|--------------------+------------+-----------+----------|
| 28.0               | 28         | 1         | F        |
| 33.666666666666664 | 101        | 3         | M        |
+--------------------+------------+-----------+----------+
```

Ejemplo 5: Calcular el máximo de un campo

El ejemplo calcula la edad máxima de todas las cuentas.

```
os> source=accounts | stats max(age);
fetched rows / total rows = 1/1
+------------+
| max(age)   |
|------------|
| 36         |
+------------+
```

Ejemplo 6: Calcular el máximo y el mínimo de un campo por grupo

El ejemplo calcula los valores de edad máxima y mínima para todas las cuentas, agrupados por género.

```
os> source=accounts | stats max(age), min(age) by gender;
fetched rows / total rows = 2/2
+------------+------------+----------+
| max(age)   | min(age)   | gender   |
|------------+------------+----------|
| 28         | 28         | F        |
| 36         | 32         | M        |
```

```
+------------+-----------+----------+
```

Ejemplo 7: Calcular el recuento distinto de un campo

Para obtener el recuento de valores distintos de un campo, puede utilizar la función `DISTINCT_COUNT` (o `DC`) en lugar de `COUNT`. El ejemplo calcula tanto el recuento como el recuento distinto del campo de género de todas las cuentas.

```
os> source=accounts | stats count(gender), distinct_count(gender);
fetched rows / total rows = 1/1
+-----------------+--------------------------+
| count(gender)   | distinct_count(gender)   |
|-----------------+--------------------------|
| 4               | 2                        |
+-----------------+--------------------------+
```

Ejemplo 8: Calcular el recuento mediante un intervalo

El ejemplo obtiene el recuento de la edad en un intervalo de 10 años.

```
os> source=accounts | stats count(age) by span(age, 10) as age_span
fetched rows / total rows = 2/2
+--------------+------------+
| count(age)   | age_span   |
|--------------+------------|
| 1            | 20         |
| 3            | 30         |
+--------------+------------+
```

Ejemplo 9: Calcule el recuento por género y rango

Este ejemplo cuenta los registros agrupados por género y edad en intervalos de 5 años.

```
os> source=accounts | stats count() as cnt by span(age, 5) as age_span, gender
fetched rows / total rows = 3/3
+-------+------------+----------+
| cnt   | age_span   | gender   |
|-------+------------+----------|
| 1     | 25         | F        |
| 2     | 30         | M        |
| 1     | 35         | M        |
+-------+------------+----------+
```

La expresión span siempre aparece como la primera clave de agrupamiento, independientemente del orden especificado en el comando.

```
os> source=accounts | stats count() as cnt by gender, span(age, 5) as age_span
fetched rows / total rows = 3/3
+-------+------------+----------+
| cnt   | age_span   | gender   |
|-------+------------+----------|
| 1     | 25         | F        |
| 2     | 30         | M        |
| 1     | 35         | M        |
+-------+------------+----------+
```

Ejemplo 10: Calcule el recuento y obtenga la lista de correo electrónico por género y rango

El ejemplo obtiene el recuento de edad por un intervalo de 10 años y el grupo por género. Además, para cada fila, obtiene una lista de 5 correos electrónicos como máximo.

```
os> source=accounts | stats count() as cnt, take(email, 5) by span(age, 5) as age_span,
 gender
fetched rows / total rows = 3/3
+-------+-----------------------------------------------------+------------+----------+
| cnt   | take(email, 5)                                      | age_span   | gender   |
|-------+-----------------------------------------------------+------------+----------|
| 1     | []                                                  | 25         | F        |
| 2     | [janedoe@anycompany.com,juanli@examplecompany.org] | 30         | M        |
| 1     | [marymajor@examplecorp.com]                         | 35         | M        |
+-------+-----------------------------------------------------+------------+----------+
```

Ejemplo 11: Calcular el percentil de un campo

El ejemplo muestra cómo calcular el percentil de 90 de edad de todas las cuentas.

```
os> source=accounts | stats percentile(age, 90);
fetched rows / total rows = 1/1
+----------------------+
| percentile(age, 90)   |
|----------------------|
| 36                   |
+----------------------+
```

Ejemplo 12: Calcular el percentil de un campo por grupo

En el ejemplo se muestra cómo calcular el percentil de 90 de edad de todas las cuentas agrupadas por género.

```
os> source=accounts | stats percentile(age, 90) by gender;
fetched rows / total rows = 2/2
+----------------------+----------+
| percentile(age, 90)   | gender   |
|----------------------+----------|
| 28                    | F        |
| 36                    | M        |
+----------------------+----------+
```

Ejemplo 13: Calcule el percentil por género y rango

El ejemplo obtiene el percentil de 90 años de edad con un intervalo de 10 años y el grupo por género.

```
os> source=accounts | stats percentile(age, 90) as p90 by span(age, 10) as age_span,
 gender
fetched rows / total rows = 2/2
+-------+------------+----------+
| p90   | age_span   | gender   |
|-------+------------+----------|
| 28    | 20         | F        |
| 36    | 30         | M        |
+-------+------------+----------+
```

```
- `source = table | stats avg(a) `
- `source = table | where a < 50 | stats avg(c) `
- `source = table | stats max(c) by b`
- `source = table | stats count(c) by b | head 5`
- `source = table | stats distinct_count(c)`
- `source = table | stats stddev_samp(c)`
- `source = table | stats stddev_pop(c)`
- `source = table | stats percentile(c, 90)`
- `source = table | stats percentile_approx(c, 99)`
```

Agregaciones con span

```
- `source = table  | stats count(a) by span(a, 10) as a_span`
- `source = table  | stats sum(age) by span(age, 5) as age_span | head 2`
- `source = table  | stats avg(age) by span(age, 20) as age_span, country  | sort -
 age_span |  head 2`
```

Agregaciones con intervalo de tiempo (función de ventana invertida)

```
- `source = table | stats sum(productsAmount) by span(transactionDate, 1d) as age_date
 | sort age_date`
- `source = table | stats sum(productsAmount) by span(transactionDate, 1w) as age_date,
 productId`
```

Las agregaciones se agrupan en varios niveles

```
- `source = table | stats avg(age) as avg_state_age by country, state | stats
 avg(avg_state_age) as avg_country_age by country`
- `source = table | stats avg(age) as avg_city_age by country, state, city | eval
 new_avg_city_age = avg_city_age - 1 | stats avg(new_avg_city_age) as avg_state_age
 by country, state | where avg_state_age > 18 | stats avg(avg_state_age) as
 avg_adult_country_age by country`
```

Comando subquery

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando PPL, consulte the section called “Comandos”.

Utilice el comando `subquery` para realizar consultas complejas y anidadas en sus sentencias del lenguaje de procesamiento canalizado (PPL).

```
source=logs | where field in [ subquery source=events | where condition | fields
 field ]
```

En este ejemplo, la búsqueda principal (`source=logs`) se filtra según los resultados de la subconsulta (`source=events`).

El comando subquery admite varios niveles de anidación para el análisis de datos complejos.

Ejemplo de subconsultas anidadas

```
source=logs | where id in [ subquery source=users | where user in [ subquery
 source=actions | where action="login" | fields user] | fields uid ]
```

Uso de InSubQuery

- `source = outer | where a in [ source = inner | fields b ]`
- `source = outer | where (a) in [ source = inner | fields b ]`
- `source = outer | where (a,b,c) in [ source = inner | fields d,e,f ]`
- `source = outer | where a not in [ source = inner | fields b ]`
- `source = outer | where (a) not in [ source = inner | fields b ]`
- `source = outer | where (a,b,c) not in [ source = inner | fields d,e,f ]`
- `source = outer a in [ source = inner | fields b ]` (filtrado de búsqueda con subconsulta)
- `source = outer a not in [ source = inner | fields b ]` (filtrado de búsqueda con subconsulta)
- `source = outer | where a in [ source = inner1 | where b not in [ source = inner2 | fields c ] | fields b ]` (anidado)
- `source = table1 | inner join left = l right = r on l.a = r.a AND r.a in [ source = inner | fields d ] | fields l.a, r.a, b, c` (como filtro de unión)

Ejemplos de migración de SQL con PPL en la subconsulta

TPC-H Q4 (subconsulta interna con agregación)

```
select
  o_orderpriority,
  count(*) as order_count
from
  orders
where
  o_orderdate >= date '1993-07-01'
  and o_orderdate < date '1993-07-01' + interval '3' month
```

```
  and o_orderkey in (
    select
      l_orderkey
    from
      lineitem
    where l_commitdate < l_receiptdate
  )
group by
  o_orderpriority
order by
  o_orderpriority
```

Reescrito por PPL en una consulta de subconsulta:

```
source = orders
| where o_orderdate >= "1993-07-01" and o_orderdate < "1993-10-01" and o_orderkey IN
  [ source = lineitem
    | where l_commitdate < l_receiptdate
    | fields l_orderkey
  ]
| stats count(1) as order_count by o_orderpriority
| sort o_orderpriority
| fields o_orderpriority, order_count
```

TPC-H Q20 (anidado en la subconsulta)

```
select
  s_name,
  s_address
from
  supplier,
  nation
where
  s_suppkey in (
    select
      ps_suppkey
    from
      partsupp
    where
      ps_partkey in (
        select
          p_partkey
        from
```

```
          part
        where
          p_name like 'forest%'
      )
  )
  and s_nationkey = n_nationkey
  and n_name = 'CANADA'
order by
  s_name
```

Reescrito por PPL en una consulta de subconsulta:

```
source = supplier
| where s_suppkey IN [
    source = partsupp
    | where ps_partkey IN [
        source = part
        | where like(p_name, "forest%")
        | fields p_partkey
      ]
    | fields ps_suppkey
  ]
| inner join left=l right=r on s_nationkey = n_nationkey and n_name = 'CANADA'
  nation
| sort s_name
```

Uso de ExistsSubQuery

Supuestos :a, b son campos del exterior de la tabla c, d son campos del interior de la tabla e, f son campos del interior de la tabla2.

- `source = outer | where exists [ source = inner | where a = c ]`
- `source = outer | where not exists [ source = inner | where a = c ]`
- `source = outer | where exists [ source = inner | where a = c and b = d ]`
- `source = outer | where not exists [ source = inner | where a = c and b = d ]`
- `source = outer exists [ source = inner | where a = c ]` (filtrado de búsqueda con subconsulta)
- `source = outer not exists [ source = inner | where a = c ]` (filtrado de búsqueda con subconsulta)

- `source = table as t1 exists [ source = table as t2 | where t1.a = t2.a ]` (el alias de la tabla es útil en una subconsulta existente)
- `source = outer | where exists [ source = inner1 | where a = c and exists [ source = inner2 | where c = e ] ]` (anidado)
- `source = outer | where exists [ source = inner1 | where a = c | where exists [ source = inner2 | where c = e ] ]` (anidado)
- `source = outer | where exists [ source = inner | where c > 10 ]` (existe una correlación no correlacionada)
- `source = outer | where not exists [ source = inner | where c > 10 ]` (existe una relación no correlacionada)
- `source = outer | where exists [ source = inner ] | eval l = "nonEmpty" | fields l` (existe una relación especial no correlacionada)

Uso de ScalarSubQuery

Supuestos: `a` `b` son campos del exterior de la tabla `c`, `d` son campos del interior de la tabla `e`, `f` son campos de la tabla anidados

Subconsulta escalar no correlacionada

Seleccione en:

- `source = outer | eval m = [ source = inner | stats max(c) ] | fields m, a`
- `source = outer | eval m = [ source = inner | stats max(c) ] + b | fields m, a`

En dónde:

- `source = outer | where a > [ source = inner | stats min(c) ] | fields a`

Búsqueda filtrada en:

- `source = outer a > [ source = inner | stats min(c) ] | fields a`

Subconsulta escalar correlacionada

Seleccione en:

- `source = outer | eval m = [ source = inner | where outer.b = inner.d | stats max(c) ] | fields m, a`
- `source = outer | eval m = [ source = inner | where b = d | stats max(c) ] | fields m, a`
- `source = outer | eval m = [ source = inner | where outer.b > inner.d | stats max(c) ] | fields m, a`

En dónde:

- `source = outer | where a = [ source = inner | where outer.b = inner.d | stats max(c) ]`
- `source = outer | where a = [ source = inner | where b = d | stats max(c) ]`
- `source = outer | where [ source = inner | where outer.b = inner.d OR inner.d = 1 | stats count() ] > 0 | fields a`

Búsqueda filtrada en:

- `source = outer a = [ source = inner | where b = d | stats max(c) ]`
- `source = outer [ source = inner | where outer.b = inner.d OR inner.d = 1 | stats count() ] > 0 | fields a`

Subconsulta escalar anidada

- `source = outer | where a = [ source = inner | stats max(c) | sort c ] OR b = [ source = inner | where c = 1 | stats min(d) | sort d ]`
- `source = outer | where a = [ source = inner | where c = [ source = nested | stats max(e) by f | sort f ] | stats max(d) by c | sort c | head 1 ]`

(Relación) Subconsulta

`InSubquery`, `ExistsSubquery` y `ScalarSubquery` son todas expresiones de subconsulta. Pero `RelationSubquery` no es una expresión de subconsulta, es un plan de subconsulta que se usa comúnmente en las cláusulas Join o From.

- `source = table1 | join left = l right = r [ source = table2 | where d > 10 | head 5 ]` (subconsulta en el lado derecho de la unión)
- `source = [ source = table1 | join left = l right = r [ source = table2 | where d > 10 | head 5 ] | stats count(a) by b ] as outer | head 1`

Contexto adicional

`InSubquery`, `ExistsSubquery` y `ScalarSubquery` son expresiones de subconsulta que se utilizan habitualmente en cláusulas `where` y filtros de búsqueda.

Comando Where:

```
| where <boolean expression> | ...
```

Filtro de búsqueda:

```
search source=* <boolean expression> | ...
```

Se puede usar una expresión de subconsulta en una expresión booleana:

```
| where orders.order_id in [ source=returns | where return_reason="damaged" | field
 order_id ]
```

El `orders.order_id in [ source=... ]` es un `<boolean expression>`.

En general, denominamos expresión `InSubquery` a este tipo de cláusula de subconsulta. Es una `<boolean expression>`.

Subconsulta con diferentes tipos de unión

Ejemplo en el que se utiliza : `ScalarSubquery`:

```
source=employees
| join source=sales on employees.employee_id = sales.employee_id
| where sales.sale_amount > [ source=targets | where target_met="true" | fields
 target_value ]
```

A diferencia de InSubquery, ExistsSubQuery y ScalarSubQuery, una RelationSubQuery no es una expresión de subconsulta. En cambio, es un plan de subconsultas.

```
SEARCH source=customer
| FIELDS c_custkey
| LEFT OUTER JOIN left = c, right = o ON c.c_custkey = o.o_custkey
   [
      SEARCH source=orders
      | WHERE o_comment NOT LIKE '%unusual%packages%'
      | FIELDS o_orderkey, o_custkey
   ]
| STATS ...
```

Comando top

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando PPL, consulte the section called “Comandos”.

Utilice el comando `top` para buscar la tupla de valores más común de todos los campos de la lista de campos.

Sintaxis

Utilice la siguiente sintaxis:

```
top [N] <field-list> [by-clause] top_approx [N] <field-list> [by-clause]
```

N

- El número de resultados que se devolverá.
- Valor predeterminado: 10

field-list

- Obligatorio.
- Una lista delimitada por comas de nombres de campo.

by-clause

- Opcional.
- Uno o más campos por los que agrupar los resultados.

top_approx

- Un recuento aproximado de los (n) campos principales mediante el uso de la cardinalidad estimada por el algoritmo HyperLogLog++.

Ejemplo 1: Encuentre los valores más comunes en un campo

En el ejemplo, se busca el género más común en todas las cuentas.

Consulta PPL:

```
os> source=accounts | top gender;
os> source=accounts | top_approx gender;
fetched rows / total rows = 2/2
+----------+
| gender   |
|----------|
| M        |
| F        |
+----------+
```

Ejemplo 2: busca los valores más comunes en un campo (limitado a 1)

En el ejemplo, se busca el género más común para todas las cuentas.

Consulta PPL:

```
os> source=accounts | top_approx 1 gender;
fetched rows / total rows = 1/1
+----------+
| gender   |
|----------|
| M        |
+----------+
```

Ejemplo 3: busca los valores más comunes agrupados por género

En el ejemplo, se busca la edad más común para todas las cuentas, agrupada por género.

Consulta PPL:

```
os> source=accounts | top 1 age by gender;
os> source=accounts | top_approx 1 age by gender;
fetched rows / total rows = 2/2
+----------+-------+
| gender   | age   |
|----------+-------|
| F        | 28    |
| M        | 32    |
+----------+-------+
```

Comando trendline

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando PPL, consulte the section called “Comandos”.

Utilice el comando `trendline` para calcular los promedios móviles de los campos.

Sintaxis

Utilice la siguiente sintaxis.

```
TRENDLINE [sort <[+|-] sort-field>] SMA(number-of-datapoints, field) [AS alias]
 [SMA(number-of-datapoints, field) [AS alias]]...
```

[+|-]

- Opcional.
- El signo más [+] indica el orden ascendente con los valores NULL o MISSING al principio.
- El signo menos [-] indica el orden descendente con los valores NULL o MISSING al final.
- Predeterminado: orden ascendente con los valores NULL o MISSING primero.

sort-field

- Obligatorio cuando se utiliza la clasificación.
- El campo que se utiliza para ordenar.

number-of-datapoints

- Obligatorio.
- El número de puntos de datos que calculan el promedio móvil.
- Debe ser mayor que cero.

campo

- Obligatorio.
- El nombre del campo para el que se debe calcular el promedio móvil.

alias

- Opcional.
- El nombre de la columna resultante que contiene el promedio móvil.

Solo se admite el tipo de promedio móvil simple (Simple Moving Average, SMA). Se calcula de la siguiente manera:

```
f[i]: The value of field 'f' in the i-th data-point
n: The number of data-points in the moving window (period)
t: The current time index

SMA(t) = (1/n) * Σ(f[i]), where i = t-n+1 to t
```

Ejemplo 1: Calcular el promedio móvil simple para una serie temporal de temperaturas

El ejemplo calcula el promedio móvil simple sobre las temperaturas utilizando dos puntos de datos.

Consulta PPL:

```
os> source=t | trendline sma(2, temperature) as temp_trend;
fetched rows / total rows = 5/5
```

```
+-----------+---------+--------------------+----------+
|temperature|device-id|           timestamp|temp_trend|
+-----------+---------+--------------------+----------+
|         12|     1492|2023-04-06 17:07:...|      NULL|
|         12|     1492|2023-04-06 17:07:...|      12.0|
|         13|      256|2023-04-06 17:07:...|      12.5|
|         14|      257|2023-04-06 17:07:...|      13.5|
|         15|      258|2023-04-06 17:07:...|      14.5|
+-----------+---------+--------------------+----------+
```

Ejemplo 2: Calcular promedios móviles simples para una serie temporal de temperaturas mediante la clasificación

El ejemplo calcula dos promedios móviles simples sobre las temperaturas utilizando dos y tres puntos de datos ordenados de forma descendente por el identificador del dispositivo.

Consulta PPL:

```
os> source=t | trendline sort - device-id sma(2, temperature) as temp_trend_2 sma(3,
 temperature) as temp_trend_3;
fetched rows / total rows = 5/5
+-----------+---------+--------------------+------------+------------------+
|temperature|device-id|           timestamp|temp_trend_2|      temp_trend_3|
+-----------+---------+--------------------+------------+------------------+
|         15|      258|2023-04-06 17:07:...|        NULL|              NULL|
|         14|      257|2023-04-06 17:07:...|        14.5|              NULL|
|         13|      256|2023-04-06 17:07:...|        13.5|              14.0|
|         12|     1492|2023-04-06 17:07:...|        12.5|              13.0|
|         12|     1492|2023-04-06 17:07:...|        12.0|12.333333333333334|
+-----------+---------+--------------------+------------+------------------+
```

Comando where

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando PPL, consulte the section called “Comandos”.

El comando `where` utiliza una expresión booleana para filtrar el resultado de la búsqueda. Solo devuelve el resultado cuando la expresión booleana se evalúa como verdadera.

Sintaxis

Utilice la siguiente sintaxis:

```
where <boolean-expression>
```

bool-expression

- Opcional.
- Cualquier expresión que pueda evaluarse en un valor booleano.

Ejemplo 1: filtrado de conjunto de resultados con condición

El ejemplo muestra cómo obtener documentos del índice de cuentas que cumplan condiciones específicas.

Consulta PPL:

```
os> source=accounts | where account_number=1 or gender="F" | fields account_number,
 gender;
fetched rows / total rows = 2/2
+------------------+----------+
| account_number   | gender   |
|------------------+----------|
| 1                | M        |
| 13               | F        |
+------------------+----------+
```

Ejemplos adicionales

Filtros con condiciones lógicas

- `source = table | where c = 'test' AND a = 1 | fields a,b,c`
- `source = table | where c != 'test' OR a > 1 | fields a,b,c | head 1`
- `source = table | where c = 'test' NOT a > 1 | fields a,b,c`
- `source = table | where a = 1 | fields a,b,c`
- `source = table | where a >= 1 | fields a,b,c`

- `source = table | where a < 1 | fields a,b,c`
- `source = table | where b != 'test' | fields a,b,c`
- `source = table | where c = 'test' | fields a,b,c | head 3`
- `source = table | where ispresent(b)`
- `source = table | where isnull(coalesce(a, b)) | fields a,b,c | head 3`
- `source = table | where isempty(a)`
- `source = table | where isblank(a)`
- `source = table | where case(length(a) > 6, 'True' else 'False') = 'True'`
- `source = table | where a between 1 and 4`, Nota: Esto devuelve un **>=** 1 y un **<=** 4, es decir, [1, 4]
- `source = table | where b not between '2024-09-10' and '2025-09-10'`, Nota: Esto devuelve b **>=** '**********' y b **<=** '2025-09-10'
- `source = table | where cidrmatch(ip, '***********/24')`
- `source = table | where cidrmatch(ipv6, '2003:db8::/32')`

```
source = table | eval status_category =
    case(a >= 200 AND a < 300, 'Success',
    a >= 300 AND a < 400, 'Redirection',
    a >= 400 AND a < 500, 'Client Error',
    a >= 500, 'Server Error'
    else 'Incorrect HTTP status code')
    | where case(a >= 200 AND a < 300, 'Success',
    a >= 300 AND a < 400, 'Redirection',
    a >= 400 AND a < 500, 'Client Error',
    a >= 500, 'Server Error'
    else 'Incorrect HTTP status code'
    ) = 'Incorrect HTTP status code'
```

```
source = table
    | eval factor = case(a > 15, a - 14, isnull(b), a - 7, a < 3, a + 1 else 1)
    | where case(factor = 2, 'even', factor = 4, 'even', factor = 6, 'even', factor =
 8, 'even' else 'odd') = 'even'
    |  stats count() by factor
```

Resumen de campo

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten este comando PPL, consulte the section called “Comandos”.

Use el comando `fieldsummary` para calcular las estadísticas básicas de cada campo (count, distinct count, min, max, avg, stddev, mean) y determine el tipo de datos de cada campo. Este comando se puede utilizar con cualquier barra vertical anterior y las tendrá en cuenta.

Sintaxis

Utilice la siguiente sintaxis. Para los casos de uso de Registros de CloudWatch, solo se admite un campo en una consulta.

```
... | fieldsummary <field-list> (nulls=true/false)
```

includefields

- Lista de todas las columnas que se van a recopilar con las estadísticas en un conjunto de resultados unificado.

Nulos

- Opcional.
- Si se establece como verdadero, incluye valores nulos en los cálculos de agregación (sustituye el valor nulo por cero para los valores numéricos).

Ejemplo 1

Consulta PPL:

```
os> source = t | where status_code != 200 | fieldsummary includefields= status_code
 nulls=true
+------------------+-------------+------------+------------+------------+------------
+------------+------------+----------------|
| Fields           | COUNT       | COUNT_DISTINCT    |  MIN  |  MAX   |  AVG   |  MEAN
  |        STDDEV       | NUlls | TYPEOF |
```

```
|-----------------+-------------+-----------+-----------+-----------+-----------
+-----------+-----------+----------------|
| "status_code"    |      2      |         2         | 301   |   403   |  352.0 |  352.0
  |  72.12489168102785  |  0    | "int"  |
+-----------------+-------------+-----------+-----------+-----------+-----------
+-----------+-----------+----------------|
```

Ejemplo 2

Consulta PPL:

```
os> source = t | fieldsummary includefields= id, status_code, request_path nulls=true
+-----------------+-------------+-----------+-----------+-----------+-----------
+-----------+-----------+----------------|
| Fields           | COUNT       | COUNT_DISTINCT    |  MIN  |  MAX   |  AVG   |  MEAN
  |        STDDEV       | NUlls | TYPEOF |
|-----------------+-------------+-----------+-----------+-----------+-----------
+-----------+-----------+----------------|
|       "id"       |      6      |         6         | 1     |   6    |  3.5   |   3.5
 |  1.8708286933869707  |  0    | "int"  |
+-----------------+-------------+-----------+-----------+-----------+-----------
+-----------+-----------+----------------|
| "status_code"    |      4      |         3         | 200   |   403   |  184.0 |  184.0
  |  161.16699413961905 |  2    | "int"  |
+-----------------+-------------+-----------+-----------+-----------+-----------
+-----------+-----------+----------------|
| "request_path"   |      2      |         2         | /about| /home   |  0.0    |  0.0
    |      0           |  2    |"string"|
+-----------------+-------------+-----------+-----------+-----------+-----------
+-----------+-----------+----------------|
```

Comando expand

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten esta función PPL, consulte the section called “Funciones”.

Utilice el comando `expand` para aplanar un campo de tipo Array<Any> o Map<Any>, lo que genera filas individuales para cada elemento o par clave-valor.

Sintaxis

Utilice la siguiente sintaxis:

```
expand <field> [As alias]
```

campo

- El campo que se va a expandir (explotar).
- El campo debe ser de un tipo compatible.

alias

- Opcional.
- El nombre que se va a utilizar en lugar del nombre del campo original.

Directrices de uso

El comando expand genera una fila para cada elemento de la matriz o campo de mapa especificado, donde:

- Los elementos de la matriz se convierten en filas individuales.
- Los pares clave-valor de la asignación se dividen en filas independientes y cada valor clave se representa como una fila.
- Cuando se proporciona un alias, los valores desglosados se representan con el alias en lugar del nombre de campo original.

Puede usar este comando en combinación con otros comandos, como stats, eval y parse, para manipular o extraer datos después de la expansión.

Ejemplos

- `source = table | expand employee | stats max(salary) as max by state, company`
- `source = table | expand employee as worker | stats max(salary) as max by state, company`

- `source = table | expand employee as worker | eval bonus = salary * 3 | fields worker, bonus`
- `source = table | expand employee | parse description '(?<email>.+@.+)' | fields employee, email`
- `source = table | eval array=json_array(1, 2, 3) | expand array as uid | fields name, occupation, uid`
- `source = table | expand multi_valueA as multiA | expand multi_valueB as multiB`

Puede usar el comando expand en combinación con otros comandos, como eval, stats y más. El uso de varios comandos de expansión creará un producto cartesiano de todos los elementos internos de cada matriz o mapa compuesto.

Consulta desplegable de SQL eficaz

El comando expand se traduce en una operación SQL equivalente mediante LATERAL VIEW explode, lo que permite la fragmentación eficiente de matrices o mapas a nivel de consulta SQL.

```
SELECT customer exploded_productId
FROM table
LATERAL VIEW explode(productId) AS exploded_productId
```

El comando explode ofrece las siguientes funciones:

- Es una operación de columna que devuelve una columna nueva.
- Crea una nueva fila para cada elemento de la columna expandida.
- Los valores nulos internos se ignoran como parte del campo fragmentado (no se crea ni explota ninguna fila en el caso de los nulos).

Funciones PPL

Temas

- Funciones de condiciones PPL
- Funciones hash criptográficas PPL
- Funciones de fecha y hora PPL
- Expresiones PPL

- Funciones PPL de dirección IP
- Nuevas funciones PPL JSON
- Funciones de Lambda PPL
- Funciones matemáticas PPL
- Funciones de cadena PPL
- Función de conversión de tipos PPL

Funciones de condiciones PPL

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten esta función PPL, consulte the section called “Funciones”.

ISNULL

Descripción: `isnull(field)` devuelve el valor verdadero si el campo es nulo.

Tipo de argumento:

- Cualquier tipo de datos compatible.

Tipo de devolución:

- BOOLEANO

Ejemplo:

```
os> source=accounts | eval result = isnull(employer) | fields result, employer,
 firstname
fetched rows / total rows = 4/4
+----------+-------------+-------------+
| result   | employer    | firstname   |
|----------+-------------+-------------|
| False    | AnyCompany  | Mary        |
| False    | ExampleCorp | Jane        |
| False    | ExampleOrg  | Nikki       |
```

```
| True     | null        | Juan        |
+----------+-------------+-------------+
```

ISNOTNULL

Descripción: `isnotnull(field)` devuelve el valor verdadero si el campo no es nulo.

Tipo de argumento:

- Cualquier tipo de datos compatible.

Tipo de devolución:

- BOOLEANO

Ejemplo:

```
os> source=accounts | where not isnotnull(employer) | fields account_number, employer
fetched rows / total rows = 1/1
+------------------+------------+
| account_number   | employer   |
|------------------+------------|
| 18               | null       |
+------------------+------------+
```

EXISTS

Ejemplo:

```
os> source=accounts | where exists(email) | fields account_number, email
fetched rows / total rows = 1/1
```

IFNULL

Descripción: `ifnull(field1, field2)` devuelve `field2` si `field1` es nulo.

Tipo de argumento:

- Cualquier tipo de datos compatible.
- Si los dos parámetros son de tipos diferentes, la función no pasará la comprobación semántica.

Tipo de devolución:

- Cualquiera

Ejemplo:

```
os> source=accounts | eval result = ifnull(employer, 'default') | fields result,
 employer, firstname
fetched rows / total rows = 4/4
+------------+------------+-------------+
| result     | employer   | firstname   |
|------------+------------+-------------|
| AnyCompany | AnyCompany | Mary        |
| ExampleCorp| ExampleCorp| Jane        |
| ExampleOrg | ExampleOrg | Nikki       |
| default    | null       | Juan        |
+------------+------------+-------------+
```

NULLIF

Descripción: `nullif(field1, field2)` devuelve nulo si dos parámetros son iguales; de lo contrario, devuelve field1.

Tipo de argumento:

- Cualquier tipo de datos compatible.
- Si los dos parámetros son de tipos diferentes, la función no pasará la comprobación semántica.

Tipo de devolución:

- Cualquiera

Ejemplo:

```
os> source=accounts | eval result = nullif(employer, 'AnyCompany') | fields result,
 employer, firstname
fetched rows / total rows = 4/4
+----------------+----------------+-------------+
| result         | employer       | firstname   |
|----------------+----------------+-------------|
```

```
| null           | AnyCompany     | Mary        |
| ExampleCorp    | ExampleCorp    | Jane        |
| ExampleOrg     | ExampleOrg     | Nikki       |
| null           | null           | Juan        |
+----------------+----------------+-------------+
```

IF

Descripción: `if(condition, expr1, expr2)` devuelve `expr1` si la condición es verdadera; de lo contrario, devuelve `expr2`.

Tipo de argumento:

- Cualquier tipo de datos compatible.
- Si los dos parámetros son de tipos diferentes, la función no pasará la comprobación semántica.

Tipo de devolución:

- Cualquiera

Ejemplo:

```
os> source=accounts | eval result = if(true, firstname, lastname) | fields result,
 firstname, lastname
fetched rows / total rows = 4/4
+----------+-------------+----------+
| result   | firstname | lastname   |
|----------+-------------+----------|
| Jane     | Jane      | Doe        |
| Mary     | Mary      | Major      |
| Pat      | Pat       | Candella   |
| Dale     | Jorge     | Souza      |
+----------+-----------+------------+

os> source=accounts | eval result = if(false, firstname, lastname) | fields result,
 firstname, lastname
fetched rows / total rows = 4/4
+----------+-------------+------------+
| result   | firstname   | lastname   |
|----------+-------------+------------|
| Doe      | Jane        | Doe        |
```

```
| Major    | Mary        | Major      |
| Candella | Pat         | Candella   |
| Souza    | Jorge       | Souza      |
+----------+-------------+------------+

os> source=accounts | eval is_vip = if(age > 30 AND isnotnull(employer), true, false) |
 fields is_vip, firstname, lastname
fetched rows / total rows = 4/4
+----------+-------------+------------+
| is_vip   | firstname   | lastname   |
|----------+-------------+------------|
| True     | Jane        | Doe        |
| True     | Mary        | Major      |
| False    | Pat         | Candella   |
| False    | Jorge       | Souza      |
+----------+-------------+------------+
```

Funciones hash criptográficas PPL

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten esta función PPL, consulte the section called “Funciones”.

MD5

MD5 calcula el resumen de MD5 y devuelve el valor en forma de cadena hexadecimal de 32 caracteres.

Uso: `md5('hello')`

Tipo de argumento:

- STRING

Tipo de devolución:

- STRING

Ejemplo:

```
os> source=people | eval `MD5('hello')` = MD5('hello') | fields `MD5('hello')`
fetched rows / total rows = 1/1
+----------------------------------+
| MD5('hello')                     |
|----------------------------------|
| <32 character hex string>        |
+----------------------------------+
```

SHA1

SHA1 devuelve el resultado de la cadena hexadecimal de SHA-1.

Uso: `sha1('hello')`

Tipo de argumento:

- STRING

Tipo de devolución:

- STRING

Ejemplo:

```
os> source=people | eval `SHA1('hello')` = SHA1('hello') | fields `SHA1('hello')`
fetched rows / total rows = 1/1
+------------------------------------------+
| SHA1('hello')                            |
|------------------------------------------|
| <40-character SHA-1 hash result>         |
+------------------------------------------+
```

SHA2

SHA2 devuelve el resultado de la cadena hexadecimal de la familia de funciones hash SHA-2 (SHA-224, SHA-256, SHA-384 y SHA-512). NumBits indica la longitud de bits deseada del resultado, que debe tener un valor de 224, 256, 384, 512

Uso:

- `sha2('hello',256)`

- `sha2('hello',512)`

Tipo de argumento:

- STRING, INTEGER

Tipo de devolución:

- STRING

Ejemplo:

```
os> source=people | eval `SHA2('hello',256)` = SHA2('hello',256) | fields
 `SHA2('hello',256)`
fetched rows / total rows = 1/1
+------------------------------------------------------------------+
| SHA2('hello',256)                                                |
|------------------------------------------------------------------|
| <64-character SHA-256 hash result>                               |
+------------------------------------------------------------------+

os> source=people | eval `SHA2('hello',512)` = SHA2('hello',512) | fields
 `SHA2('hello',512)`
fetched rows / total rows = 1/1
+------------------------------------------------------------------+
| SHA2('hello',512)                                                |
                                          |
|------------------------------------------------------------------|
| <128-character SHA-512 hash result>                              |
+------------------------------------------------------------------+
```

Funciones de fecha y hora PPL

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten esta función PPL, consulte the section called “Funciones”.

DAY

Uso: DAY(date) extrae el día del mes para una fecha, en el rango del 1 al 31.

Tipo de argumento: STRING/DATE/TIMESTAMP

Tipo de devolución: INTEGER

Sinónimos: DAYOFMONTH, DAY_OF_MONTH

Ejemplo:

```
os> source=people | eval `DAY(DATE('2020-08-26'))` = DAY(DATE('2020-08-26')) | fields
 `DAY(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
+---------------------------+
| DAY(DATE('2020-08-26'))   |
|---------------------------|
| 26                        |
+---------------------------+
```

DAYOFMONTH

Uso: DAYOFMONTH(date) extrae el día del mes para una fecha, en el rango del 1 al 31.

Tipo de argumento: STRING/DATE/TIMESTAMP

Tipo de devolución: INTEGER

Sinónimos: DAY, DAY_OF_MONTH

Ejemplo:

```
os> source=people | eval `DAYOFMONTH(DATE('2020-08-26'))` =
 DAYOFMONTH(DATE('2020-08-26')) | fields `DAYOFMONTH(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
+---------------------------------+
| DAYOFMONTH(DATE('2020-08-26'))   |
|---------------------------------|
| 26                              |
+---------------------------------+
```

DAY_OF_MONTH

Uso: DAY_OF_MONTH(DATE) extrae el día del mes para una fecha, en el rango del 1 al 31.

Tipo de argumento: STRING/DATE/TIMESTAMP

Tipo de devolución: INTEGER

Sinónimos: DAY, DAYOFMONTH

Ejemplo:

```
os> source=people | eval `DAY_OF_MONTH(DATE('2020-08-26'))` =
 DAY_OF_MONTH(DATE('2020-08-26')) | fields `DAY_OF_MONTH(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
+-----------------------------------+
| DAY_OF_MONTH(DATE('2020-08-26'))   |
|-----------------------------------|
| 26                                |
+-----------------------------------+
```

DAYOFWEEK

Uso: DAYOFWEEK(DATE) devuelve el índice de los días de la semana de una fecha (1 = domingo, 2 = lunes,..., 7 = sábado).

Tipo de argumento: STRING/DATE/TIMESTAMP

Tipo de devolución: INTEGER

Sinónimos: DAY_OF_WEEK

Ejemplo:

```
os> source=people | eval `DAYOFWEEK(DATE('2020-08-26'))` =
 DAYOFWEEK(DATE('2020-08-26')) | fields `DAYOFWEEK(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
+---------------------------------+
| DAYOFWEEK(DATE('2020-08-26'))   |
|---------------------------------|
| 4                               |
+---------------------------------+
```

DAY_OF_WEEK

Uso: DAY_OF_WEEK(DATE) devuelve el índice de los días de la semana de una fecha (1 = domingo, 2 = lunes,..., 7 = sábado).

Tipo de argumento: STRING/DATE/TIMESTAMP

Tipo de devolución: INTEGER

Sinónimos: DAYOFWEEK

Ejemplo:

```
os> source=people | eval `DAY_OF_WEEK(DATE('2020-08-26'))` =
 DAY_OF_WEEK(DATE('2020-08-26')) | fields `DAY_OF_WEEK(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
+-----------------------------------+
| DAY_OF_WEEK(DATE('2020-08-26'))   |
|-----------------------------------|
| 4                                 |
+-----------------------------------+
```

DAYOFYEAR

Uso: DAYOFYEAR(DATE) devuelve el día del año de una fecha, en el rango de 1 a 366.

Tipo de argumento: STRING/DATE/TIMESTAMP

Tipo de devolución: INTEGER

Sinónimos: DAY_OF_YEAR

Ejemplo:

```
os> source=people | eval `DAYOFYEAR(DATE('2020-08-26'))` =
 DAYOFYEAR(DATE('2020-08-26')) | fields `DAYOFYEAR(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
+-------------------------------+
| DAYOFYEAR(DATE('2020-08-26'))   |
|-------------------------------|
| 239                           |
```

```
+--------------------------------+
```

DAY_OF_YEAR

Uso: DAY_OF_YEAR(DATE) devuelve el día del año de una fecha, en el rango de 1 a 366.

Tipo de argumento: STRING/DATE/TIMESTAMP

Tipo de devolución: INTEGER

Sinónimos: DAYOFYEAR

Ejemplo:

```
os> source=people | eval `DAY_OF_YEAR(DATE('2020-08-26'))` =
 DAY_OF_YEAR(DATE('2020-08-26')) | fields `DAY_OF_YEAR(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
+----------------------------------+
| DAY_OF_YEAR(DATE('2020-08-26'))  |
|----------------------------------|
| 239                              |
+----------------------------------+
```

DAYNAME

Uso: DAYNAME(DATE) devuelve el nombre del día de la semana para una fecha, incluidos lunes, martes, miércoles, jueves, viernes, sábado y domingo.

Tipo de argumento: STRING/DATE/TIMESTAMP

Tipo de devolución: STRING

Ejemplo:

```
os> source=people | eval `DAYNAME(DATE('2020-08-26'))` = DAYNAME(DATE('2020-08-26')) |
 fields `DAYNAME(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
+------------------------------+
| DAYNAME(DATE('2020-08-26'))  |
|------------------------------|
| Wednesday                    |
+------------------------------+
```

FROM_UNIXTIME

Uso: FROM_UNIXTIME devuelve una representación del argumento dado como marca de tiempo o valor de cadena de caracteres. Esta función realiza una conversión inversa de la función UNIX_TIMESTAMP.

Si proporciona un segundo argumento, FROM_UNIXTIME lo utiliza para dar al resultado un formato similar al de la función DATE_FORMAT.

Si la marca de tiempo está fuera del intervalo 1970-01-01 00:00:00 a 3001-01-18 23:59:59.999999 (0 a 32536771199.999999 tiempo de época), la función devuelve NULL.

Tipo de argumento: DOUBLE, STRING

Mapa del tipo de devolución:

DOUBLE -> TIMESTAMP

DOUBLE, STRING -> STRING

Ejemplos:

```
os> source=people | eval `FROM_UNIXTIME(1220249547)` = FROM_UNIXTIME(1220249547) |
 fields `FROM_UNIXTIME(1220249547)`
fetched rows / total rows = 1/1
+-----------------------------+
| FROM_UNIXTIME(1220249547)   |
|-----------------------------|
| 2008-09-01 06:12:27         |
+-----------------------------+

os> source=people | eval `FROM_UNIXTIME(1220249547, 'HH:mm:ss')` =
 FROM_UNIXTIME(1220249547, 'HH:mm:ss') | fields `FROM_UNIXTIME(1220249547, 'HH:mm:ss')`
fetched rows / total rows = 1/1
+---------------------------------------+
| FROM_UNIXTIME(1220249547, 'HH:mm:ss')   |
|---------------------------------------|
| 06:12:27                              |
+---------------------------------------+
```

HOUR

Uso: HOUR(TIME) extrae el valor hora del tiempo.

A diferencia de una hora del día estándar, el valor de hora de esta función puede tener un rango superior a 23. Como resultado, el valor devuelto de `HOUR(TIME)` puede ser superior a 23.

Tipo de argumento: STRING/TIME/TIMESTAMP

Tipo de devolución: INTEGER

Sinónimos: `HOUR_OF_DAY`

Ejemplo:

```
os> source=people | eval `HOUR(TIME('01:02:03'))` = HOUR(TIME('01:02:03')) | fields
 `HOUR(TIME('01:02:03'))`
fetched rows / total rows = 1/1
+-------------------------+
| HOUR(TIME('01:02:03'))   |
|-------------------------|
| 1                       |
+-------------------------+
```

`HOUR_OF_DAY`

Uso: `HOUR_OF_DAY(TIME)` extrae el valor de la hora del tiempo dado.

A diferencia de una hora del día estándar, el valor de hora de esta función puede tener un rango superior a 23. Como resultado, el valor devuelto de `HOUR_OF_DAY(TIME)` puede ser superior a 23.

Tipo de argumento: STRING/TIME/TIMESTAMP

Tipo de devolución: INTEGER

Sinónimos: `HOUR`

Ejemplo:

```
os> source=people | eval `HOUR_OF_DAY(TIME('01:02:03'))` =
 HOUR_OF_DAY(TIME('01:02:03')) | fields `HOUR_OF_DAY(TIME('01:02:03'))`
fetched rows / total rows = 1/1
+--------------------------------+
| HOUR_OF_DAY(TIME('01:02:03'))   |
|--------------------------------|
| 1                              |
```

```
+--------------------------------+
```

LAST_DAY

Uso: LAST_DAY devuelve el último día del mes como un valor de fecha para el argumento de fecha indicado.

Tipo de argumento: DATE/STRING/TIMESTAMP/TIME

Tipo de devolución: DATE

Ejemplo:

```
os> source=people | eval `last_day('2023-02-06')` = last_day('2023-02-06') | fields
 `last_day('2023-02-06')`
fetched rows / total rows = 1/1
+--------------------------+
| last_day('2023-02-06')   |
|--------------------------|
| 2023-02-28               |
+--------------------------+
```

LOCALTIMESTAMP

Uso: LOCALTIMESTAMP() es un sinónimo de NOW().

Ejemplo:

```
> source=people | eval `LOCALTIMESTAMP()` = LOCALTIMESTAMP() | fields
 `LOCALTIMESTAMP()`
fetched rows / total rows = 1/1
+---------------------+
| LOCALTIMESTAMP()    |
|---------------------|
| 2022-08-02 15:54:19 |
+---------------------+
```

LOCALTIME

Uso: LOCALTIME() es un sinónimo de NOW().

Ejemplo:

```
> source=people | eval `LOCALTIME()` = LOCALTIME() | fields `LOCALTIME()`
fetched rows / total rows = 1/1
+---------------------+
| LOCALTIME()         |
|---------------------|
| 2022-08-02 15:54:19 |
+---------------------+
```

MAKE_DATE

Uso: MAKE_DATE devuelve un valor de fecha basado en los valores de año, mes y día dados. Todos los argumentos se redondean a números enteros.

Especificaciones: 1. MAKE_DATE(INTEGER, INTEGER, INTEGER) -> DATE

Tipo de argumento: INTEGER, INTEGER, INTEGER

Tipo de devolución: DATE

Ejemplo:

```
os> source=people | eval `MAKE_DATE(1945, 5, 9)` = MAKEDATE(1945, 5, 9) | fields
 `MAKEDATE(1945, 5, 9)`
fetched rows / total rows = 1/1
+------------------------+
| MAKEDATE(1945, 5, 9)   |
|------------------------|
| 1945-05-09             |
+------------------------+
```

MINUTE

Uso: MINUTE(TIME) devuelve el componente de minutos del tiempo dado, como un número entero en el rango de 0 a 59.

Tipo de argumento: STRING/TIME/TIMESTAMP

Tipo de devolución: INTEGER

Sinónimos: MINUTE_OF_HOUR

Ejemplo:

```
os> source=people | eval `MINUTE(TIME('01:02:03'))` =  MINUTE(TIME('01:02:03')) |
 fields `MINUTE(TIME('01:02:03'))`
fetched rows / total rows = 1/1
+----------------------------+
| MINUTE(TIME('01:02:03'))   |
|----------------------------|
| 2                          |
+----------------------------+
```

MINUTE_OF_HOUR

Uso: `MINUTE_OF_HOUR(TIME)` devuelve el componente de minutos del tiempo dado, como un número entero en el rango de 0 a 59.

Tipo de argumento: STRING/TIME/TIMESTAMP

Tipo de devolución: INTEGER

Sinónimos: `MINUTE`

Ejemplo:

```
os> source=people | eval `MINUTE_OF_HOUR(TIME('01:02:03'))` =
 MINUTE_OF_HOUR(TIME('01:02:03')) | fields `MINUTE_OF_HOUR(TIME('01:02:03'))`
fetched rows / total rows = 1/1
+-----------------------------------+
| MINUTE_OF_HOUR(TIME('01:02:03'))   |
|-----------------------------------|
| 2                                 |
+-----------------------------------+
```

MONTH

Uso: `MONTH(DATE)` devuelve el mes de la fecha dada como un entero, en el rango de 1 a 12 (donde 1 representa enero y 12 representa diciembre).

Tipo de argumento: STRING/DATE/TIMESTAMP

Tipo de devolución: INTEGER

Sinónimos: `MONTH_OF_YEAR`

Ejemplo:

```
os> source=people | eval `MONTH(DATE('2020-08-26'))` =  MONTH(DATE('2020-08-26')) |
 fields `MONTH(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
+-----------------------------+
| MONTH(DATE('2020-08-26'))   |
|-----------------------------|
| 8                           |
+-----------------------------+
```

MONTHNAME

Uso: `MONTHNAME(DATE)` devuelve el mes de la fecha dada como un entero, en el rango de 1 a 12 (donde 1 representa enero y 12 representa diciembre).

Tipo de argumento: STRING/DATE/TIMESTAMP

Tipo de devolución: INTEGER

Sinónimos: `MONTH_OF_YEAR`

Ejemplo:

```
os> source=people | eval `MONTHNAME(DATE('2020-08-26'))` =
 MONTHNAME(DATE('2020-08-26')) | fields `MONTHNAME(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
+--------------------------------+
| MONTHNAME(DATE('2020-08-26'))   |
|--------------------------------|
| August                         |
+--------------------------------+
```

MONTH_OF_YEAR

Uso: `MONTH_OF_YEAR(DATE)` devuelve el mes de la fecha dada como un entero, en el rango de 1 a 12 (donde 1 representa enero y 12 representa diciembre).

Tipo de argumento: STRING/DATE/TIMESTAMP

Tipo de devolución: INTEGER

Sinónimos: `MONTH`

Ejemplo:

```
os> source=people | eval `MONTH_OF_YEAR(DATE('2020-08-26'))` =
 MONTH_OF_YEAR(DATE('2020-08-26')) | fields `MONTH_OF_YEAR(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
+-------------------------------------+
| MONTH_OF_YEAR(DATE('2020-08-26'))   |
|-------------------------------------|
| 8                                   |
+-------------------------------------+
```

NOW

Uso: NOW devuelve la fecha y la hora actuales como un valor TIMESTAMP en el formato 'YYYY-MM-DD hh:mm:ss' . El valor se expresa en la zona horaria del clúster.

> **Note**
>
> NOW( ) devuelve un tiempo constante que indica cuándo comenzó a ejecutarse la sentencia. Esto es diferente de SYSDATE( ), que devuelve la hora exacta de ejecución.

Tipo de devolución: TIMESTAMP

Especificación: NOW () -> TIMESTAMP

Ejemplo:

```
os> source=people | eval `value_1` = NOW(), `value_2` = NOW() | fields `value_1`,
 `value_2`
fetched rows / total rows = 1/1
+---------------------+---------------------+
| value_1             | value_2             |
|---------------------+---------------------|
| 2022-08-02 15:39:05 | 2022-08-02 15:39:05 |
+---------------------+---------------------+
```

QUARTER

Uso: QUARTER(DATE) devuelve el trimestre del año de la fecha dada como un entero, en el rango de 1 a 4.

Tipo de argumento: STRING/DATE/TIMESTAMP

Tipo de devolución: INTEGER

Ejemplo:

```
os> source=people | eval `QUARTER(DATE('2020-08-26'))` = QUARTER(DATE('2020-08-26')) |
 fields `QUARTER(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
+-------------------------------+
| QUARTER(DATE('2020-08-26'))   |
|-------------------------------|
| 3                             |
+-------------------------------+
```

SECOND

Uso: `SECOND(TIME)` devuelve el segundo componente del tiempo dado como un entero, en el rango de 0 a 59.

Tipo de argumento: STRING/TIME/TIMESTAMP

Tipo de devolución: INTEGER

Sinónimos: `SECOND_OF_MINUTE`

Ejemplo:

```
os> source=people | eval `SECOND(TIME('01:02:03'))` = SECOND(TIME('01:02:03')) | fields
 `SECOND(TIME('01:02:03'))`
fetched rows / total rows = 1/1
+---------------------------+
| SECOND(TIME('01:02:03'))   |
|---------------------------|
| 3                         |
+---------------------------+
```

SECOND_OF_MINUTE

Uso: `SECOND_OF_MINUTE(TIME)` devuelve el segundo componente del tiempo dado como un entero, en el rango de 0 a 59.

Tipo de argumento: STRING/TIME/TIMESTAMP

Tipo de devolución: INTEGER

Sinónimos: SECOND

Ejemplo:

```
os> source=people | eval `SECOND_OF_MINUTE(TIME('01:02:03'))` =
 SECOND_OF_MINUTE(TIME('01:02:03')) | fields `SECOND_OF_MINUTE(TIME('01:02:03'))`
fetched rows / total rows = 1/1
+-------------------------------------+
| SECOND_OF_MINUTE(TIME('01:02:03'))  |
|-------------------------------------|
| 3                                   |
+-------------------------------------+
```

SUBDATE

Uso: SUBDATE(DATE, DAYS) resta el segundo argumento (como DATE o DAYS) de la fecha dada.

Tipo de argumento: DATE/TIMESTAMP, LONG

Mapa de tipos de devolución: (DATE, LONG) -> DATE

Antónimos: ADDDATE

Ejemplo:

```
os> source=people | eval `'2008-01-02' - 31d` = SUBDATE(DATE('2008-01-02'), 31),
 `'2020-08-26' - 1` = SUBDATE(DATE('2020-08-26'), 1), `ts '2020-08-26 01:01:01' -
 1` = SUBDATE(TIMESTAMP('2020-08-26 01:01:01'), 1) | fields `'2008-01-02' - 31d`,
 `'2020-08-26' - 1`, `ts '2020-08-26 01:01:01' - 1`
fetched rows / total rows = 1/1
+---------------------+--------------------+-------------------------------+
| '2008-01-02' - 31d  | '2020-08-26' - 1   | ts '2020-08-26 01:01:01' - 1  |
|---------------------+--------------------+-------------------------------|
| 2007-12-02 00:00:00 | 2020-08-25         | 2020-08-25 01:01:01           |
+---------------------+--------------------+-------------------------------+
```

SYSDATE

Uso: SYSDATE() devuelve la fecha y la hora actuales como un valor TIMESTAMP en el formato 'YYYY-MM-DD hh:mm:ss.nnnnnn'.

SYSDATE() devuelve la hora exacta en la que se ejecuta. Esto difiere de NOW(), que devuelve un tiempo constante que indica cuándo comenzó a ejecutarse la sentencia.

Tipo de argumento opcional: INTEGER (0 a 6): especifica el número de dígitos para las fracciones de segundo del valor devuelto.

Tipo de devolución: TIMESTAMP

Ejemplo:

```
os> source=people | eval `SYSDATE()` = SYSDATE() | fields `SYSDATE()`
fetched rows / total rows = 1/1
+----------------------------+
| SYSDATE()                  |
|----------------------------|
| 2022-08-02 15:39:05.123456 |
+----------------------------+
```

`TIMESTAMP`

Uso: `TIMESTAMP(EXPR)` construye un tipo de marca de tiempo con la cadena `expr` de entrada como marca de tiempo.

Con un solo argumento, `TIMESTAMP(expr)` construye una marca de tiempo a partir de la entrada. Si `expr` es una cadena, se interpreta como una marca de tiempo. Para los argumentos que no son cadenas, la función convierte `expr` en una marca de tiempo utilizando la zona horaria UTC. Cuando `expr` es un valor `TIME`, la función aplica la fecha de hoy antes de la conversión.

Cuando se usa con dos argumentos, `TIMESTAMP(expr1, expr2)` agrega la expresión de hora (`expr2`) a la expresión de fecha o marca de tiempo (`expr1`) y devuelve el resultado como un valor de marca de tiempo.

Tipo de argumento: STRING/DATE/TIME/TIMESTAMP

Mapa del tipo de devolución:

(STRING/DATE/TIME/TIMESTAMP) -> TIMESTAMP

(STRING/DATE/TIME/TIMESTAMP, STRING/DATE/TIME/TIMESTAMP) -> TIMESTAMP

Ejemplo:

```
os> source=people | eval `TIMESTAMP('2020-08-26 13:49:00')` = TIMESTAMP('2020-08-26
 13:49:00'), `TIMESTAMP('2020-08-26 13:49:00', TIME('12:15:42'))` =
 TIMESTAMP('2020-08-26 13:49:00', TIME('12:15:42')) | fields `TIMESTAMP('2020-08-26
 13:49:00')`, `TIMESTAMP('2020-08-26 13:49:00', TIME('12:15:42'))`
```

```
fetched rows / total rows = 1/1
+-----------------------------------
+-----------------------------------------------------+
| TIMESTAMP('2020-08-26 13:49:00')   | TIMESTAMP('2020-08-26 13:49:00',
 TIME('12:15:42'))   |
|-----------------------------------
+-----------------------------------------------------|
| 2020-08-26 13:49:00               | 2020-08-27 02:04:42
     |
+-----------------------------------
+-----------------------------------------------------+
```

UNIX_TIMESTAMP

Uso: UNIX_TIMESTAMP convierte un argumento de fecha determinado a la hora de Unix (segundos desde la Época, que comenzó a principios de 1970). Si no se proporciona ningún argumento, devuelve la hora actual de Unix.

El argumento de fecha puede ser un DATE, una cadena TIMESTAMP o un número en uno de estos formatos: YYMMDD, YYMMDDhhmmss, YYYYMMDD, o YYYYMMDDhhmmss. Si el argumento incluye un componente de tiempo, puede incluir opcionalmente fracciones de segundo.

Si el formato del argumento no es válido o se encuentra fuera del intervalo comprendido entre 1970-01-01 00:00:00 a 3001-01-18 23:59:59.999999 (0 a 32536771199.999999 en tiempo de época), la función devuelve NULL.

La función acepta DATE, TIMESTAMP, o DOUBLE como tipos de argumentos o no acepta ningún argumento. Siempre devuelve un valor DOUBLE que representa la marca de tiempo de Unix.

Para la conversión inversa, puede utilizar la función FROM_UNIXTIME.

Tipo de argumento: <NONE>/DOUBLE/DATE/TIMESTAMP

Tipo de devolución: DOUBLE

Ejemplo:

```
os> source=people | eval `UNIX_TIMESTAMP(double)` = UNIX_TIMESTAMP(20771122143845),
 `UNIX_TIMESTAMP(timestamp)` = UNIX_TIMESTAMP(TIMESTAMP('1996-11-15 17:05:42')) |
 fields `UNIX_TIMESTAMP(double)`, `UNIX_TIMESTAMP(timestamp)`
fetched rows / total rows = 1/1
+-------------------------+---------------------------+
| UNIX_TIMESTAMP(double)   | UNIX_TIMESTAMP(timestamp)   |
```

```
|--------------------------+-----------------------------|
| 3404817525.0             | 848077542.0                 |
+--------------------------+-----------------------------+
```

WEEK

Uso: `WEEK(DATE)` devuelve el número de semana de una fecha determinada.

Tipo de argumento: DATE/TIMESTAMP/STRING

Tipo de devolución: INTEGER

Sinónimos: `WEEK_OF_YEAR`

Ejemplo:

```
os> source=people | eval `WEEK(DATE('2008-02-20'))` = WEEK(DATE('2008-02-20')) | fields
 `WEEK(DATE('2008-02-20'))`
fetched rows / total rows = 1/1
+----------------------------+
| WEEK(DATE('2008-02-20'))   |
|----------------------------|
| 8                          |
+----------------------------+
```

WEEKDAY

Uso: `WEEKDAY(DATE)` devuelve el índice del día de la semana correspondiente a la fecha (0 = lunes, 1 = martes,..., 6 = domingo).

Es similar a la función `dayofweek`, pero devuelve índices diferentes para cada día.

Tipo de argumento: STRING/DATE/TIME/TIMESTAMP

Tipo de devolución: INTEGER

Ejemplo:

```
os> source=people | eval `weekday(DATE('2020-08-26'))` = weekday(DATE('2020-08-26'))
 | eval `weekday(DATE('2020-08-27'))` = weekday(DATE('2020-08-27')) | fields
 `weekday(DATE('2020-08-26'))`, `weekday(DATE('2020-08-27'))`
fetched rows / total rows = 1/1
+------------------------------+------------------------------+
| weekday(DATE('2020-08-26'))  | weekday(DATE('2020-08-27'))  |
```

```
|------------------------------+------------------------------|
| 2                            | 3                            |
+------------------------------+------------------------------+
```

WEEK_OF_YEAR

Uso: `WEEK_OF_YEAR(DATE)` devuelve el número de la semana de la fecha indicada.

Tipo de argumento: DATE/TIMESTAMP/STRING

Tipo de devolución: INTEGER

Sinónimos: `WEEK`

Ejemplo:

```
os> source=people | eval `WEEK_OF_YEAR(DATE('2008-02-20'))` = WEEK(DATE('2008-02-20'))|
 fields `WEEK_OF_YEAR(DATE('2008-02-20'))`
fetched rows / total rows = 1/1
+-----------------------------------+
| WEEK_OF_YEAR(DATE('2008-02-20'))  |
|-----------------------------------|
| 8                                 |
+-----------------------------------+
```

YEAR

Uso: `YEAR(DATE)` devuelve el año para la fecha, en el rango de 1000 a 9999, o 0 para la fecha “zero”.

Tipo de argumento: STRING/DATE/TIMESTAMP

Tipo de devolución: INTEGER

Ejemplo:

```
os> source=people | eval `YEAR(DATE('2020-08-26'))` = YEAR(DATE('2020-08-26')) | fields
 `YEAR(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
+---------------------------+
| YEAR(DATE('2020-08-26'))  |
|---------------------------|
| 2020                      |
```

```
+---------------------------+
```

DATE_ADD

Uso: `DATE_ADD(date, INTERVAL expr unit)` agrega el intervalo especificado a la fecha dada.

Tipo de argumento: DATE, INTERVAL

Tipo de devolución: DATE

Antónimos: `DATE_SUB`

Ejemplo:

```
os> source=people | eval `'2020-08-26' + 1d` = DATE_ADD(DATE('2020-08-26'), INTERVAL 1
 DAY) | fields `'2020-08-26' + 1d`
fetched rows / total rows = 1/1
+---------------------+
| '2020-08-26' + 1d   |
|---------------------|
| 2020-08-27          |
+---------------------+
```

DATE_SUB

Uso: `DATE_SUB(date, INTERVAL expr unit)` resta el intervalo de expiración de la fecha.

Tipo de argumento: DATE, INTERVAL

Tipo de devolución: DATE

Antónimos: `DATE_ADD`

Ejemplo:

```
os> source=people | eval `'2008-01-02' - 31d` = DATE_SUB(DATE('2008-01-02'), INTERVAL
 31 DAY) | fields `'2008-01-02' - 31d`
fetched rows / total rows = 1/1
+---------------------+
| '2008-01-02' - 31d  |
|---------------------|
| 2007-12-02          |
+---------------------+
```

TIMESTAMPADD

Uso: devuelve un valor TIMESTAMP después de agregar un intervalo de tiempo específico a una fecha determinada.

Argumentos:

- intervalo: INTERVAL (SECOND, MINUTE, HOUR, DAY, WEEK, MONTH, QUARTER, YEAR)
- entero: INTEGER
- fecha: DATE, TIMESTAMP o STRING

Si proporciona un argumento de fecha STRING, formatéelo como válido TIMESTAMP. La función convierte automáticamente un argumento DATE en un TIMESTAMP.

Ejemplos:

```
os> source=people | eval `TIMESTAMPADD(DAY, 17, '2000-01-01 00:00:00')` =
 TIMESTAMPADD(DAY, 17, '2000-01-01 00:00:00') | eval `TIMESTAMPADD(QUARTER, -1,
 '2000-01-01 00:00:00')` = TIMESTAMPADD(QUARTER, -1, '2000-01-01 00:00:00') | fields
 `TIMESTAMPADD(DAY, 17, '2000-01-01 00:00:00')`, `TIMESTAMPADD(QUARTER, -1, '2000-01-01
 00:00:00')`
fetched rows / total rows = 1/1
+-----------------------------------------------
+-------------------------------------------------+
| TIMESTAMPADD(DAY, 17, '2000-01-01 00:00:00') | TIMESTAMPADD(QUARTER, -1, '2000-01-01
 00:00:00') |
|-----------------------------------------------
+-------------------------------------------------|
| 2000-01-18 00:00:00                          | 1999-10-01 00:00:00
           |
+-----------------------------------------------
+-------------------------------------------------+
```

TIMESTAMPDIFF

Uso: TIMESTAMPDIFF(interval, start, end) devuelve la diferencia entre la fecha/hora de inicio y finalización en unidades de intervalo especificadas.

Argumentos:

- intervalo: INTERVAL (SECOND, MINUTE, HOUR, DAY, WEEK, MONTH, QUARTER, YEAR)
- inicio: DATE, TIMESTAMP, o STRING
- final: DATE, TIMESTAMP, o STRING

La función convierte automáticamente los argumentos `TIMESTAMP` cuando sea apropiado. Formatee los argumentos `STRING` como válidos `TIMESTAMP`.

Ejemplos:

```
os> source=people | eval `TIMESTAMPDIFF(YEAR, '1997-01-01 00:00:00', '2001-03-06
 00:00:00')` = TIMESTAMPDIFF(YEAR, '1997-01-01 00:00:00', '2001-03-06 00:00:00') |
 eval `TIMESTAMPDIFF(SECOND, timestamp('1997-01-01 00:00:23'), timestamp('1997-01-01
 00:00:00'))` = TIMESTAMPDIFF(SECOND, timestamp('1997-01-01 00:00:23'),
 timestamp('1997-01-01 00:00:00')) | fields `TIMESTAMPDIFF(YEAR, '1997-01-01 00:00:00',
 '2001-03-06 00:00:00')`, `TIMESTAMPDIFF(SECOND, timestamp('1997-01-01 00:00:23'),
 timestamp('1997-01-01 00:00:00'))`
fetched rows / total rows = 1/1
+-------------------------------------------------------------------
+-------------------------------------------------------------------------------------------
+
| TIMESTAMPDIFF(YEAR, '1997-01-01 00:00:00', '2001-03-06 00:00:00') |
 TIMESTAMPDIFF(SECOND, timestamp('1997-01-01 00:00:23'), timestamp('1997-01-01
 00:00:00')) |
|-------------------------------------------------------------------
+-------------------------------------------------------------------------------------------|
| 4                                                                 | -23
                                                                        |
+-------------------------------------------------------------------
+-------------------------------------------------------------------------------------------
+
```

`UTC_TIMESTAMP`

Uso: `UTC_TIMESTAMP` devuelve la marca de tiempo UTC actual como un valor en 'YYYY-MM-DD hh:mm:ss'.

Tipo de devolución: TIMESTAMP

Especificación: UTC_TIMESTAMP() -> TIMESTAMP

Ejemplo:

```
> source=people | eval `UTC_TIMESTAMP()` = UTC_TIMESTAMP() | fields `UTC_TIMESTAMP()`
fetched rows / total rows = 1/1
+---------------------+
| UTC_TIMESTAMP()     |
|---------------------|
| 2022-10-03 17:54:28 |
+---------------------+
```

CURRENT_TIMEZONE

Uso: CURRENT_TIMEZONE devuelve la zona horaria local actual.

Tipo de devolución: STRING

Ejemplo:

```
> source=people | eval `CURRENT_TIMEZONE()` = CURRENT_TIMEZONE() | fields
 `CURRENT_TIMEZONE()`
fetched rows / total rows = 1/1
+------------------------+
| CURRENT_TIMEZONE()     |
|------------------------|
| America/Chicago        |
+------------------------+
```

Expresiones PPL

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten esta función PPL, consulte the section called “Funciones”.

Las expresiones, en particular las expresiones de valores, devuelven un valor escalar. Las expresiones tienen distintos tipos y formas. Por ejemplo, hay valores literales como expresiones atómicas y expresiones aritméticas, de predicados y de funciones construidas sobre ellos. Puede utilizar expresiones en distintas cláusulas, como las expresiones aritméticas en los comandos `Filter` y `Stats`.

Operadores

Una expresión aritmética es una expresión formada por literales numéricos y operadores aritméticos binarios, de la siguiente manera:

1. +: suma.
2. -: resta.
3. *: multiplicación.
4. /: división (en el caso de los números enteros, el resultado es un número entero sin la parte fraccionaria)
5. %: módulo (se usa solo con números enteros; el resultado es el resto de la división)

Prioridad

Use paréntesis para controlar la precedencia de los operadores aritméticos. De lo contrario, los operadores de mayor prioridad se ejecutan primero.

Conversión de tipos

La conversión de tipos implícita se realiza al buscar las firmas de los operadores. Por ejemplo, en un entero + un número real coincide con la firma `+(double,double)`, lo que da como resultado un número real. Esta regla también se aplica a las llamadas de funciones.

Ejemplo de diferentes tipos de expresiones aritméticas:

```
os> source=accounts | where age > (25 + 5) | fields age ;
fetched rows / total rows = 3/3
+-------+
| age   |
|-------|
| 32    |
| 36    |
| 33    |
+-------+
```

Operadores de predicados

Un operador de predicado es una expresión que se evalúa como verdadera. La comparación entre el valor `MISSING` y el valor `NULL` sigue estas reglas:

- Un valor `MISSING` solo es igual a un valor `MISSING` y es menor que otros valores.

- Un valor `NULL` es igual a un valor `NULL`, es mayor que un valor `MISSING`, pero menor que todos los demás valores.

Operadores

Operadores de predicados

Nombre	Descripción
`>`	El operador mayor que
`>=`	Operador Mayor o igual que
`<`	El operador menor que
`!=`	Operador no igual
`<=`	Los operadores menor que o igual a
`=`	Operador igual
`LIKE`	Coincidencia de patrones simple
`IN`	Prueba de valor NULL
`AND`	Operador AND
`OR`	Operador OR
`XOR`	Operador XOR
`NOT`	Prueba de valor NOT NULL

Puede comparar fechas y horas. Al comparar diferentes tipos de fecha y hora (por ejemplo, `DATE` y `TIME`), ambos se convierten en `DATETIME`. Las siguientes reglas se aplican para la conversión:

- `TIME` se aplica a la fecha de hoy.
- `DATE` se interpreta a medianoche.

Operador de predicado básico

Ejemplo de operadores de comparación:

```
os> source=accounts | where age > 33 | fields age ;
fetched rows / total rows = 1/1
+-------+
| age   |
|-------|
| 36    |
+-------+
```

IN

Ejemplo del campo de prueba `IN` del operador en las listas de valores:

```
os> source=accounts | where age in (32, 33) | fields age ;
fetched rows / total rows = 2/2
+-------+
| age   |
|-------|
| 32    |
| 33    |
+-------+
```

OR

Ejemplo del operador `OR`:

```
os> source=accounts | where age = 32 OR age = 33 | fields age ;
fetched rows / total rows = 2/2
+-------+
| age   |
|-------|
| 32    |
| 33    |
+-------+
```

NOT

Ejemplo del operador `NOT`:

```
os> source=accounts | where age not in (32, 33) | fields age ;
fetched rows / total rows = 2/2
```

```
+-------+
| age   |
|-------|
| 36    |
| 28    |
+-------+
```

Funciones PPL de dirección IP

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten esta función PPL, consulte the section called “Funciones”.

`CIDRMATCH`

Uso: `CIDRMATCH(ip, cidr)` comprueba si la dirección IP especificada está dentro del rango de cidr dado.

Tipo de argumento:

- STRING, STRING
- Tipo de devolución: BOOLEAN

Ejemplo:

```
os> source=ips | where cidrmatch(ip, '***********/24') | fields ip
fetched rows / total rows = 1/1
+--------------+
| ip           |
|--------------|
| ***********  |
+--------------+

os> source=ipsv6 | where cidrmatch(ip, '2003:db8::/32') | fields ip
fetched rows / total rows = 1/1
+-----------------------------------------+
| ip                                      |
|-----------------------------------------|
| 2003:0db8:****:****:****:****:****:0000 |
```

```
+----------------------------------------+
```

> **Note**
>
> - `ip` puede ser una dirección IPv4 o IPv6.
> - `cidr` puede ser un bloque de IPv4 o IPv6.
> - `ip` y `cidr` debe ser tanto IPv4 como IPv6 o ambas.
> - `ip` y `cidr` deben ser ambos válidos y no estar vacíos ni ser nulos.

Nuevas funciones PPL JSON

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten esta función PPL, consulte the section called “Funciones”.

`JSON`

Uso: `json(value)` evalúa si una cadena se puede analizar en formato JSON. La función devuelve la cadena original si es un JSON válido o nula si no es válido.

Tipo de argumento: STRING

Tipo de devolución: STRING/NULL. Una expresión STRING de un formato de objeto JSON válido.

Ejemplos:

```
os> source=people | eval `valid_json()` = json('[1,2,3,{"f1":1,"f2":[5,6]},4]') |
 fields valid_json
fetched rows / total rows = 1/1
+----------------------------------+
| valid_json                       |
+----------------------------------+
| [1,2,3,{"f1":1,"f2":[5,6]},4]    |
+----------------------------------+

os> source=people | eval `invalid_json()` = json('{"invalid": "json"') | fields
 invalid_json
```

```
fetched rows / total rows = 1/1
+----------------+
| invalid_json   |
+----------------+
| null           |
+----------------+
```

JSON_OBJECT

Uso: `json_object(<key>, <value>[, <key>, <value>]...)` devuelve un objeto JSON de los miembros de pares clave-valor.

Tipo de argumento:

- La **<key>** debe ser STRING.
- El **<value>** puede ser cualquier tipo de dato.

Tipo de devolución: JSON_OBJECT. Una expresión StructType de un objeto JSON válido.

Ejemplos:

```
os> source=people | eval result = json_object('key', 123.45) | fields result
fetched rows / total rows = 1/1
+------------------+
| result           |
+------------------+
| {"key":123.45}   |
+------------------+

os> source=people | eval result = json_object('outer', json_object('inner', 123.45)) |
 fields result
fetched rows / total rows = 1/1
+-----------------------------+
| result                      |
+-----------------------------+
| {"outer":{"inner":123.45}}  |
+-----------------------------+
```

JSON_ARRAY

Uso: `json_array(<value>...)` crea una matriz JSON usando una lista de valores.

Tipo de argumento: un <value> puede ser cualquier tipo de valor, como una cadena, un número o un booleano.

Tipo de devolución: ARRAY. Una matriz de cualquier tipo de datos compatible para una matriz JSON válida.

Ejemplos:

```
os> source=people | eval `json_array` = json_array(1, 2, 0, -1, 1.1, -0.11)
fetched rows / total rows = 1/1
+-----------------------------+
| json_array                  |
+-----------------------------+
| [1.0,2.0,0.0,-1.0,1.1,-0.11] |
+-----------------------------+

os> source=people | eval `json_array_object` = json_object("array", json_array(1, 2, 0,
 -1, 1.1, -0.11))
fetched rows / total rows = 1/1
+---------------------------------------+
| json_array_object                     |
+---------------------------------------+
| {"array":[1.0,2.0,0.0,-1.0,1.1,-0.11]} |
+---------------------------------------+
```

TO_JSON_STRING

Uso: to_json_string(jsonObject) devuelve una cadena JSON con un valor de objeto json determinado.

Tipo de argumento: JSON_OBJECT

Tipo de devolución: STRING

Ejemplos:

```
os> source=people | eval `json_string` = to_json_string(json_array(1, 2, 0, -1, 1.1,
 -0.11)) | fields json_string
fetched rows / total rows = 1/1
+-------------------------------+
| json_string                   |
+-------------------------------+
| [1.0,2.0,0.0,-1.0,1.1,-0.11]   |
```

```
+--------------------------------+

os> source=people | eval `json_string` = to_json_string(json_object('key', 123.45)) |
 fields json_string
fetched rows / total rows = 1/1
+-----------------+
| json_string     |
+-----------------+
| {'key', 123.45} |
+-----------------+
```

ARRAY_LENGTH

Uso: `array_length(jsonArray)` devuelve la cantidad de elementos en la matriz exterior.

Tipo de argumento: ARRAY. Un objeto ARRAY o JSON_ARRAY.

Tipo de devolución: INTEGER

Ejemplo:

```
os> source=people | eval `json_array` = json_array_length(json_array(1,2,3,4)),
 `empty_array` = json_array_length(json_array())
fetched rows / total rows = 1/1
+--------------+---------------+
| json_array   | empty_array   |
+--------------+---------------+
| 4            | 0             |
+--------------+---------------+
```

JSON_EXTRACT

Uso: `json_extract(jsonStr, path)` extrae un objeto JSON de una cadena JSON en función de la ruta JSON especificada. La función devuelve un valor nulo si la cadena JSON de entrada no es válida.

Tipo de argumento: STRING, STRING

Tipo de devolución: STRING

- Una expresión STRING de un formato de objeto JSON válido.
- NULL se devuelve en caso de un JSON no válido.

Ejemplos:

```
os> source=people | eval `json_extract('{"a":"b"}', '$.a')` = json_extract('{"a":"b"}',
 '$a')
fetched rows / total rows = 1/1
+----------------------------------+
| json_extract('{"a":"b"}', 'a')   |
+----------------------------------+
| b                                |
+----------------------------------+

os> source=people | eval `json_extract('{"a":[{"b":1},{"b":2}]}', '$.a[1].b')` =
 json_extract('{"a":[{"b":1},{"b":2}]}', '$.a[1].b')
fetched rows / total rows = 1/1
+------------------------------------------------------------+
| json_extract('{"a":[{"b":1.0},{"b":2.0}]}', '$.a[1].b')   |
+------------------------------------------------------------+
| 2.0                                                        |
+------------------------------------------------------------+

os> source=people | eval `json_extract('{"a":[{"b":1},{"b":2}]}', '$.a[*].b')` =
 json_extract('{"a":[{"b":1},{"b":2}]}', '$.a[*].b')
fetched rows / total rows = 1/1
+------------------------------------------------------------+
| json_extract('{"a":[{"b":1.0},{"b":2.0}]}', '$.a[*].b')   |
+------------------------------------------------------------+
| [1.0,2.0]                                                  |
+------------------------------------------------------------+

os> source=people | eval `invalid_json` = json_extract('{"invalid": "json"')
fetched rows / total rows = 1/1
+----------------+
| invalid_json   |
+----------------+
| null           |
+----------------+
```

JSON_KEYS

Uso: `json_keys(jsonStr)` devuelve todas las claves del objeto JSON más externo como una matriz.

Tipo de argumento: STRING. Una expresión STRING de un formato de objeto JSON válido.

Tipo de devolución: ARRAY[STRING]. La función devuelve NULL para cualquier otra cadena JSON válida, una cadena vacía o un JSON no válido.

Ejemplos:

```
os> source=people | eval `keys` = json_keys('{"f1":"abc","f2":{"f3":"a","f4":"b"}}')
fetched rows / total rows = 1/1
+------------+
| keus       |
+------------+
| [f1, f2]   |
+------------+

os> source=people | eval `keys` = json_keys('[1,2,3,{"f1":1,"f2":[5,6]},4]')
fetched rows / total rows = 1/1
+--------+
| keys   |
+--------+
| null   |
+--------+
```

JSON_VALID

Uso: `json_valid(jsonStr)` evalúa si una cadena JSON usa una sintaxis JSON válida y devuelve TRUE o FALSE.

Tipo de argumento: STRING

Tipo de devolución: BOOLEAN

Ejemplos:

```
os> source=people | eval `valid_json` = json_valid('[1,2,3,4]'), `invalid_json` =
 json_valid('{"invalid": "json"') | feilds `valid_json`, `invalid_json`
fetched rows / total rows = 1/1
+--------------+----------------+
| valid_json   | invalid_json   |
+--------------+----------------+
| True         | False          |
+--------------+----------------+

os> source=accounts | where json_valid('[1,2,3,4]') and isnull(email) | fields
 account_number, email
```

```
fetched rows / total rows = 1/1
+------------------+---------+
| account_number   | email   |
|------------------+---------|
| 13               | null    |
+------------------+---------+
```

Funciones de Lambda PPL

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten esta función PPL, consulte the section called “Funciones”.

`EXISTS`

Uso: `exists(array, lambda)` evalúa si un predicado de Lambda es válido para uno o más elementos de la matriz.

Tipo de argumento: ARRAY, LAMBDA

Tipo de devolución: BOOLEAN. Devuelve `TRUE` si al menos un elemento de la matriz satisface el predicado Lambda; en caso contrario `FALSE`.

Ejemplos:

```
 os> source=people | eval array = json_array(1, -1, 2), result = exists(array, x -> x >
 0) | fields result
fetched rows / total rows = 1/1
+-----------+
| result    |
+-----------+
| true      |
+-----------+

os> source=people | eval array = json_array(-1, -3, -2), result = exists(array, x -> x
 > 0) | fields result
fetched rows / total rows = 1/1
+-----------+
| result    |
+-----------+
```

```
| false     |
+-----------+
```

FILTER

Uso: `filter(array, lambda)` filtra la matriz de entrada utilizando la función de Lambda dada.

Tipo de argumento: ARRAY, LAMBDA

Tipo de devolución: ARRAY. Un ARRAY que contiene todos los elementos de la matriz de entrada que satisfacen el predicado lambda.

Ejemplos:

```
 os> source=people | eval array = json_array(1, -1, 2), result = filter(array, x -> x >
 0) | fields result
fetched rows / total rows = 1/1
+-----------+
| result    |
+-----------+
| [1, 2]    |
+-----------+

os> source=people | eval array = json_array(-1, -3, -2), result = filter(array, x -> x
 > 0) | fields result
fetched rows / total rows = 1/1
+-----------+
| result    |
+-----------+
| []        |
+-----------+
```

TRANSFORM

Uso: `transform(array, lambda)` transforma los elementos de una matriz mediante la función de transformación de Lambda. El segundo argumento implica el índice del elemento si se utiliza la función de Lambda binaria. Esto es similar a lo que ocurre a `map` en la programación funcional.

Tipo de argumento: ARRAY, LAMBDA

Tipo de devolución: ARRAY. Un ARRAY que contiene el resultado de aplicar la función de transformación de Lambda a cada elemento de la matriz de entrada.

Ejemplos:

```
os> source=people | eval array = json_array(1, 2, 3), result = transform(array, x -> x
 + 1) | fields result
fetched rows / total rows = 1/1
+--------------+
| result       |
+--------------+
| [2, 3, 4]    |
+--------------+

os> source=people | eval array = json_array(1, 2, 3), result = transform(array, (x, i)
 -> x + i) | fields result
fetched rows / total rows = 1/1
+--------------+
| result       |
+--------------+
| [1, 3, 5]    |
+--------------+
```

REDUCE

Uso: `reduce(array, start, merge_lambda, finish_lambda)` reduce una matriz a un único valor mediante la aplicación de funciones de Lambda. La función aplica la combinación merge_lambda al valor inicial y a todos los elementos de la matriz y, a continuación, la aplica al resultado `finish_lambda`.

Tipo de argumento: ARRAY, ANY, LAMBDA, LAMBDA

Tipo de devolución: ANY. El resultado final de aplicar las funciones de Lambda al valor inicial y a la matriz de entrada.

Ejemplos:

```
 os> source=people | eval array = json_array(1, 2, 3), result = reduce(array, 0, (acc,
 x) -> acc + x) | fields result
fetched rows / total rows = 1/1
+-----------+
| result    |
+-----------+
| 6         |
+-----------+
```

```
os> source=people | eval array = json_array(1, 2, 3), result = reduce(array, 10, (acc,
 x) -> acc + x) | fields result
fetched rows / total rows = 1/1
+-----------+
| result    |
+-----------+
| 16        |
+-----------+

os> source=people | eval array = json_array(1, 2, 3), result = reduce(array, 0, (acc,
 x) -> acc + x, acc -> acc * 10) | fields result
fetched rows / total rows = 1/1
+-----------+
| result    |
+-----------+
| 60        |
+-----------+
```

Funciones matemáticas PPL

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten esta función PPL, consulte the section called “Funciones”.

ABS

Uso: `ABS(x)` calcula el valor absoluto de x.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de devolución: INTEGER/LONG/FLOAT/DOUBLE

Ejemplo:

```
os> source=people | eval `ABS(-1)` = ABS(-1) | fields `ABS(-1)`
fetched rows / total rows = 1/1
+-----------+
| ABS(-1)   |
|-----------|
```

```
| 1         |
+-----------+
```

ACOS

Uso: `ACOS(x)` calcula el arco coseno de x. Retorna `NULL` si x no está en el rango de -1 a 1.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de devolución: DOUBLE

Ejemplo:

```
os> source=people | eval `ACOS(0)` = ACOS(0) | fields `ACOS(0)`
fetched rows / total rows = 1/1
+--------------------+
| ACOS(0)            |
|--------------------|
| 1.5707963267948966 |
+--------------------+
```

ASIN

Uso: `asin(x)` calcula el arco sinusoidal de x. Retorna `NULL` si x no está en el rango de -1 a 1.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de devolución: DOUBLE

Ejemplo:

```
os> source=people | eval `ASIN(0)` = ASIN(0) | fields `ASIN(0)`
fetched rows / total rows = 1/1
+-----------+
| ASIN(0)   |
|-----------|
| 0.0       |
+-----------+
```

ATAN

Uso: `ATAN(x)` calcula el arco tangente de x. `atan(y, x)` Calcula el arco tangente de y/x, excepto que los signos de ambos argumentos determinan el cuadrante del resultado.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de devolución: DOUBLE

Ejemplo:

```
os> source=people | eval `ATAN(2)` = ATAN(2), `ATAN(2, 3)` = ATAN(2, 3) | fields
 `ATAN(2)`, `ATAN(2, 3)`
fetched rows / total rows = 1/1
+--------------------+--------------------+
| ATAN(2)            | ATAN(2, 3)         |
|--------------------+--------------------|
| 1.1071487177940904 | 0.5880026035475675 |
+--------------------+--------------------+
```

ATAN2

Uso: `ATAN2(y, x)` calcula el arco tangente de y/x, excepto que los signos de ambos argumentos determinan el cuadrante del resultado.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de devolución: DOUBLE

Ejemplo:

```
os> source=people | eval `ATAN2(2, 3)` = ATAN2(2, 3) | fields `ATAN2(2, 3)`
fetched rows / total rows = 1/1
+--------------------+
| ATAN2(2, 3)        |
|--------------------|
| 0.5880026035475675 |
+--------------------+
```

CBRT

Uso: CBRT calcula la raíz cuadrada de un número.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de devolución: DOUBLE:

INTEGER/LONG/FLOAT/DOUBLE -> DOUBLE

Ejemplo:

```
opensearchsql> source=location | eval `CBRT(8)` = CBRT(8), `CBRT(9.261)` = CBRT(9.261),
 `CBRT(-27)` = CBRT(-27) | fields `CBRT(8)`, `CBRT(9.261)`, `CBRT(-27)`;
fetched rows / total rows = 2/2
+-----------+---------------+-------------+
| CBRT(8)   | CBRT(9.261)   | CBRT(-27)   |
|-----------+---------------+-------------|
| 2.0       | 2.1           | -3.0        |
| 2.0       | 2.1           | -3.0        |
+-----------+---------------+-------------+
```

CEIL

Uso: Un alias para la función `CEILING`. `CEILING(T)` toma el límite máximo del valor T.

Limitación: `CEILING` solo funciona según lo esperado cuando el tipo doble IEEE 754 muestra un decimal al guardarlo.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de devolución: LONG

Ejemplo:

```
os> source=people | eval `CEILING(0)` = CEILING(0), `CEILING(50.00005)` =
 CEILING(50.00005), `CEILING(-50.00005)` = CEILING(-50.00005) | fields `CEILING(0)`,
 `CEILING(50.00005)`, `CEILING(-50.00005)`
fetched rows / total rows = 1/1
+--------------+---------------------+----------------------+
| CEILING(0)   | CEILING(50.00005)   | CEILING(-50.00005)   |
|--------------+---------------------+----------------------|
| 0            | 51                  | -50                  |
+--------------+---------------------+----------------------+

os> source=people | eval `CEILING(3147483647.12345)` = CEILING(3147483647.12345),
 `CEILING(113147483647.12345)` = CEILING(113147483647.12345),
 `CEILING(3147483647.00001)` = CEILING(3147483647.00001) | fields
 `CEILING(3147483647.12345)`, `CEILING(113147483647.12345)`,
 `CEILING(3147483647.00001)`
fetched rows / total rows = 1/1
+----------------------------+------------------------------
+----------------------------+
```

```
| CEILING(3147483647.12345)   | CEILING(113147483647.12345)   |
 CEILING(3147483647.00001)   |
|-----------------------------+-------------------------------
+-----------------------------|
| 3147483648                  | 113147483648                  | 3147483648
     |
+-----------------------------+-------------------------------
+-----------------------------+
```

CONV

Uso: `CONV(x, a, b)` convierte el número x de base a a una base b.

Tipo de argumento: x: STRING, a: INTEGER, b: INTEGER

Tipo de devolución: STRING

Ejemplo:

```
os> source=people | eval `CONV('12', 10, 16)` = CONV('12', 10, 16), `CONV('2C', 16,
 10)` = CONV('2C', 16, 10), `CONV(12, 10, 2)` = CONV(12, 10, 2), `CONV(1111, 2, 10)` =
 CONV(1111, 2, 10) | fields `CONV('12', 10, 16)`, `CONV('2C', 16, 10)`, `CONV(12, 10,
 2)`, `CONV(1111, 2, 10)`
fetched rows / total rows = 1/1
+----------------------+----------------------+-------------------
+----------------------+
| CONV('12', 10, 16)   | CONV('2C', 16, 10)   | CONV(12, 10, 2)   | CONV(1111, 2, 10)
 |
|----------------------+----------------------+-------------------
+----------------------|
| c                    | 44                   | 1100              | 15
 |
+----------------------+----------------------+-------------------
+----------------------+
```

COS

Uso: `COS(x)` calcula el coseno de x, donde x se expresa en radianes.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de devolución: DOUBLE

Ejemplo:

```
os> source=people | eval `COS(0)` = COS(0) | fields `COS(0)`
fetched rows / total rows = 1/1
+----------+
| COS(0)   |
|----------|
| 1.0      |
+----------+
```

COT

Uso: `COT(x)` calcula la cotangente de x. Devuelve un error de fuera de intervalo si x es igual a 0.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de devolución: DOUBLE

Ejemplo:

```
os> source=people | eval `COT(1)` = COT(1) | fields `COT(1)`
fetched rows / total rows = 1/1
+--------------------+
| COT(1)             |
|--------------------|
| 0.6420926159343306 |
+--------------------+
```

CRC32

Uso: `CRC32` calcula un valor de comprobación de redundancia cíclica y devuelve un valor sin signo de 32 bits.

Tipo de argumento: STRING

Tipo de devolución: LONG

Ejemplo:

```
os> source=people | eval `CRC32('MySQL')` = CRC32('MySQL') | fields `CRC32('MySQL')`
fetched rows / total rows = 1/1
+------------------+
| CRC32('MySQL')   |
|------------------|
```

```
| 3259397556       |
+------------------+
```

DEGREES

Uso: `DEGREES(x)` convierte x de radianes a grados.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de devolución: DOUBLE

Ejemplo:

```
os> source=people | eval `DEGREES(1.57)` = DEGREES(1.57) | fields `DEGREES(1.57)`
fetched rows / total rows  = 1/1
+------------------+
| DEGREES(1.57)    |
|------------------|
| 89.95437383553924 |
+------------------+
```

E

Uso: `E()` devuelve el número de Euler.

Tipo de devolución: DOUBLE

Ejemplo:

```
os> source=people | eval `E()` = E() | fields `E()`
fetched rows / total rows = 1/1
+------------------+
| E()              |
|------------------|
| 2.718281828459045 |
+------------------+
```

EXP

Uso: `EXP(x)` devuelve e elevado a x.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de devolución: DOUBLE

Ejemplo:

```
os> source=people | eval `EXP(2)` = EXP(2) | fields `EXP(2)`
fetched rows / total rows = 1/1
+------------------+
| EXP(2)           |
|------------------|
| 7.38905609893065 |
+------------------+
```

FLOOR

Uso: FLOOR(T) toma el mínimo del valor T.

Limitación: FLOOR solo funciona según lo esperado cuando el tipo doble IEEE 754 muestra un decimal cuando se almacena.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de devolución: LONG

Ejemplo:

```
os> source=people | eval `FLOOR(0)` = FLOOR(0), `FLOOR(50.00005)` = FLOOR(50.00005),
 `FLOOR(-50.00005)` = FLOOR(-50.00005) | fields `FLOOR(0)`, `FLOOR(50.00005)`,
 `FLOOR(-50.00005)`
fetched rows / total rows = 1/1
+------------+------------------+--------------------+
| FLOOR(0)   | FLOOR(50.00005)   | FLOOR(-50.00005)   |
|------------+------------------+--------------------|
| 0          | 50                | -51                |
+------------+------------------+--------------------+

os> source=people | eval `FLOOR(3147483647.12345)` = FLOOR(3147483647.12345),
 `FLOOR(113147483647.12345)` = FLOOR(113147483647.12345), `FLOOR(3147483647.00001)`
 = FLOOR(3147483647.00001) | fields `FLOOR(3147483647.12345)`,
 `FLOOR(113147483647.12345)`, `FLOOR(3147483647.00001)`
fetched rows / total rows = 1/1
+---------------------------+-----------------------------+---------------------------+
| FLOOR(3147483647.12345)   | FLOOR(113147483647.12345)   | FLOOR(3147483647.00001)   |
|---------------------------+-----------------------------+---------------------------|
```

```
| 3147483647                | 113147483647                | 3147483647                |
+---------------------------+-----------------------------+---------------------------+

os> source=people | eval `FLOOR(282474973688888.022)` = FLOOR(282474973688888.022),
 `FLOOR(9223372036854775807.022)` = FLOOR(9223372036854775807.022),
 `FLOOR(9223372036854775807.0000001)` = FLOOR(9223372036854775807.0000001)
 | fields `FLOOR(282474973688888.022)`, `FLOOR(9223372036854775807.022)`,
 `FLOOR(9223372036854775807.0000001)`
fetched rows / total rows = 1/1
+-----------------------------+-----------------------------------
+-------------------------------------+
| FLOOR(282474973688888.022)   | FLOOR(9223372036854775807.022)   |
 FLOOR(9223372036854775807.0000001)   |
|-----------------------------+-----------------------------------
+-------------------------------------|
| 282474973688888              | 9223372036854775807              | 9223372036854775807
                  |
+-----------------------------+-----------------------------------
+-------------------------------------+
```

LN

Uso: `LN(x)` devuelve el logaritmo natural de x.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de devolución: DOUBLE

Ejemplo:

```
os> source=people | eval `LN(2)` = LN(2) | fields `LN(2)`
fetched rows / total rows = 1/1
+--------------------+
| LN(2)              |
|--------------------|
| 0.6931471805599453 |
+--------------------+
```

LOG

Uso: `LOG(x)` devuelve el logaritmo natural de x que es el logaritmo en base e de x. log(B, x) es equivalente a log(x)/log(B).

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de devolución: DOUBLE

Ejemplo:

```
os> source=people | eval `LOG(2)` = LOG(2), `LOG(2, 8)` = LOG(2, 8) | fields `LOG(2)`,
 `LOG(2, 8)`
fetched rows / total rows = 1/1
+--------------------+-------------+
| LOG(2)             | LOG(2, 8)   |
|--------------------+-------------|
| 0.6931471805599453 | 3.0         |
+--------------------+-------------+
```

LOG2

Uso: `LOG2(x)` equivale a `log(x)/log(2)`.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de devolución: DOUBLE

Ejemplo:

```
os> source=people | eval `LOG2(8)` = LOG2(8) | fields `LOG2(8)`
fetched rows / total rows = 1/1
+-----------+
| LOG2(8)   |
|-----------|
| 3.0       |
+-----------+
```

LOG10

Uso: `LOG10(x)` equivale a `log(x)/log(10)`.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de devolución: DOUBLE

Ejemplo:

```
os> source=people | eval `LOG10(100)` = LOG10(100) | fields `LOG10(100)`
fetched rows / total rows = 1/1
+--------------+
| LOG10(100)   |
|--------------|
| 2.0          |
+--------------+
```

MOD

Uso: `MOD(n, m)` calcula el resto del número n dividido por m.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de devolución: Tipo más amplio entre los tipos de n y m si m es un valor distinto de cero. Si m es igual a 0, devuelve NULL.

Ejemplo:

```
os> source=people | eval `MOD(3, 2)` = MOD(3, 2), `MOD(3.1, 2)` = MOD(3.1, 2) | fields
 `MOD(3, 2)`, `MOD(3.1, 2)`
fetched rows / total rows = 1/1
+-------------+---------------+
| MOD(3, 2)   | MOD(3.1, 2)   |
|-------------+---------------|
| 1           | 1.1           |
+-------------+---------------+
```

PI

Uso: `PI()` devuelve la constante pi.

Tipo de devolución: DOUBLE

Ejemplo:

```
os> source=people | eval `PI()` = PI() | fields `PI()`
fetched rows / total rows = 1/1
+-------------------+
| PI()              |
|-------------------|
| 3.141592653589793 |
```

```
+-------------------+
```

POW

Uso: POW(x, y) calcula el valor de x elevado a la potencia de y. Las entradas incorrectas devuelven un resultado NULL.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de devolución: DOUBLE

Sinónimos: POWER(_, _)

Ejemplo:

```
os> source=people | eval `POW(3, 2)` = POW(3, 2), `POW(-3, 2)` = POW(-3, 2), `POW(3,
 -2)` = POW(3, -2) | fields `POW(3, 2)`, `POW(-3, 2)`, `POW(3, -2)`
fetched rows / total rows = 1/1
+-------------+--------------+--------------------+
| POW(3, 2)   | POW(-3, 2)   | POW(3, -2)         |
|-------------+--------------+--------------------|
| 9.0         | 9.0          | 0.1111111111111111 |
+-------------+--------------+--------------------+
```

POWER

Uso: POWER(x, y) calcula el valor de x elevado a la potencia de y. Las entradas incorrectas devuelven un resultado NULL.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de devolución: DOUBLE

Sinónimos: POW(_, _)

Ejemplo:

```
os> source=people | eval `POWER(3, 2)` = POWER(3, 2), `POWER(-3, 2)` = POWER(-3, 2),
 `POWER(3, -2)` = POWER(3, -2) | fields `POWER(3, 2)`, `POWER(-3, 2)`, `POWER(3, -2)`
fetched rows / total rows = 1/1
+---------------+----------------+--------------------+
| POWER(3, 2)   | POWER(-3, 2)   | POWER(3, -2)       |
```

```
|---------------+----------------+--------------------|
| 9.0           | 9.0            | 0.1111111111111111 |
+---------------+----------------+--------------------+
```

RADIANS

Uso: `RADIANS(x)` convierte x de grados a radianes.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de devolución: DOUBLE

Ejemplo:

```
os> source=people | eval `RADIANS(90)` = RADIANS(90) | fields `RADIANS(90)`
fetched rows / total rows  = 1/1
+--------------------+
| RADIANS(90)        |
|--------------------|
| 1.5707963267948966 |
+--------------------+
```

RAND

Uso: `RAND()`/`RAND(N)` devuelve un valor de punto flotante aleatorio en el rango 0 <= value < 1,0. Si especifica el entero N, la función inicializa la semilla antes de la ejecución. Una consecuencia de este comportamiento es que, con un argumento N idéntico, `rand(N)` devuelve el mismo valor cada vez, lo que produce una secuencia repetible de valores de columna.

Tipo de argumento: INTEGER

Tipo de devolución: FLOAT

Ejemplo:

```
os> source=people | eval `RAND(3)` = RAND(3) | fields `RAND(3)`
fetched rows / total rows = 1/1
+------------+
| RAND(3)    |
|------------|
| 0.73105735 |
+------------+
```

ROUND

Uso: `ROUND(x, d)` redondea el argumento de x a d decimales. Si no especifica d, el valor predeterminado es 0.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Mapa del tipo de devolución:

- (INTEGER/LONG [,INTEGER]) -> LONG
- (FLOAT/DOUBLE [,INTEGER]) -> LONG

Ejemplo:

```
os> source=people | eval `ROUND(12.34)` = ROUND(12.34), `ROUND(12.34, 1)` =
 ROUND(12.34, 1), `ROUND(12.34, -1)` = ROUND(12.34, -1), `ROUND(12, 1)` = ROUND(12, 1)
 | fields `ROUND(12.34)`, `ROUND(12.34, 1)`, `ROUND(12.34, -1)`, `ROUND(12, 1)`
fetched rows / total rows = 1/1
+----------------+-------------------+--------------------+----------------+
| ROUND(12.34)   | ROUND(12.34, 1)   | ROUND(12.34, -1)   | ROUND(12, 1)   |
|----------------+-------------------+--------------------+----------------|
| 12.0           | 12.3              | 10.0               | 12             |
+----------------+-------------------+--------------------+----------------+
```

SIGN

Uso: `SIGN` devuelve el signo del argumento como -1, 0 o 1, dependiendo de si el número es negativo, cero o positivo.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de devolución: INTEGER

Ejemplo:

```
os> source=people | eval `SIGN(1)` = SIGN(1), `SIGN(0)` = SIGN(0), `SIGN(-1.1)` =
 SIGN(-1.1) | fields `SIGN(1)`, `SIGN(0)`, `SIGN(-1.1)`
fetched rows / total rows = 1/1
+-----------+-----------+--------------+
| SIGN(1)   | SIGN(0)   | SIGN(-1.1)   |
|-----------+-----------+--------------|
| 1         | 0         | -1           |
```

```
+-----------+-----------+--------------+
```

SIN

Uso: `sin(x)` calcula el seno de x, donde x se expresa en radianes.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de devolución: DOUBLE

Ejemplo:

```
os> source=people | eval `SIN(0)` = SIN(0) | fields `SIN(0)`
fetched rows / total rows = 1/1
+----------+
| SIN(0)   |
|----------|
| 0.0      |
+----------+
```

SQRT

Uso: SQRT calcula la raíz cuadrada de un número no negativo.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Mapa del tipo de devolución:

- (No negativo) INTEGER/LONG/FLOAT/DOUBLE -> DOUBLE
- (Negativo) INTEGER/LONG/FLOAT/DOUBLE -> NULL

Ejemplo:

```
os> source=people | eval `SQRT(4)` = SQRT(4), `SQRT(4.41)` = SQRT(4.41) | fields
 `SQRT(4)`, `SQRT(4.41)`
fetched rows / total rows = 1/1
+-----------+--------------+
| SQRT(4)   | SQRT(4.41)   |
|-----------+--------------|
| 2.0       | 2.1          |
+-----------+--------------+
```

Funciones de cadena PPL

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten esta función PPL, consulte the section called “Funciones”.

CONCAT

Uso: `CONCAT(str1, str2, ...., str_9)` suma hasta 9 cadenas.

Tipo de argumento:

- STRING, STRING, ..., STRING
- Tipo de devolución: STRING

Ejemplo:

```
os> source=people | eval `CONCAT('hello', 'world')` = CONCAT('hello', 'world'),
 `CONCAT('hello ', 'whole ', 'world', '!')` = CONCAT('hello ', 'whole ', 'world', '!')
 | fields `CONCAT('hello', 'world')`, `CONCAT('hello ', 'whole ', 'world', '!')`
fetched rows / total rows = 1/1
+----------------------------+----------------------------------------------+
| CONCAT('hello', 'world')   | CONCAT('hello ', 'whole ', 'world', '!')   |
|----------------------------+----------------------------------------------|
| helloworld                 | hello whole world!                           |
+----------------------------+----------------------------------------------+
```

CONCAT_WS

Uso: `CONCAT_WS(sep, str1, str2)` concatena dos o más cadenas utilizando un separador específico entre ellas.

Tipo de argumento:

- STRING, STRING, ..., STRING
- Tipo de devolución: STRING

Ejemplo:

```
os> source=people | eval `CONCAT_WS(',', 'hello', 'world')` = CONCAT_WS(',', 'hello',
 'world') | fields `CONCAT_WS(',', 'hello', 'world')`
fetched rows / total rows = 1/1
+-----------------------------------+
| CONCAT_WS(',', 'hello', 'world')   |
|-----------------------------------|
| hello,world                       |
+-----------------------------------+
```

LENGTH

Uso: `length(str)` devuelve la longitud de la cadena de entrada medida en bytes.

Tipo de argumento:

- STRING
- Tipo de devolución: INTEGER

Ejemplo:

```
os> source=people | eval `LENGTH('helloworld')` = LENGTH('helloworld') | fields
 `LENGTH('helloworld')`
fetched rows / total rows = 1/1
+------------------------+
| LENGTH('helloworld')   |
|------------------------|
| 10                     |
+------------------------+
```

LOWER

Uso: `lower(string)` convierte la cadena de caracteres en minúsculas.

Tipo de argumento:

- STRING
- Tipo de devolución: STRING

Ejemplo:

```
os> source=people | eval `LOWER('helloworld')` = LOWER('helloworld'),
 `LOWER('HELLOWORLD')` = LOWER('HELLOWORLD') | fields `LOWER('helloworld')`,
 `LOWER('HELLOWORLD')`
fetched rows / total rows = 1/1
+-----------------------+-----------------------+
| LOWER('helloworld')   | LOWER('HELLOWORLD')   |
|-----------------------+-----------------------|
| helloworld            | helloworld            |
+-----------------------+-----------------------+
```

LTRIM

Uso: `ltrim(str)` elimina los espacios iniciales de la cadena de entrada.

Tipo de argumento:

- STRING
- Tipo de devolución: STRING

Ejemplo:

```
os> source=people | eval `LTRIM('   hello')` = LTRIM('   hello'), `LTRIM('hello   ')` =
 LTRIM('hello   ') | fields `LTRIM('   hello')`, `LTRIM('hello   ')`
fetched rows / total rows = 1/1
+---------------------+---------------------+
| LTRIM('   hello')   | LTRIM('hello   ')   |
|---------------------+---------------------|
| hello               | hello               |
+---------------------+---------------------+
```

POSITION

Uso: `POSITION(substr IN str)` devuelve la posición de la primera aparición de la subcadena en la cadena. Devuelve 0 si no se encuentra la subcadena dentro de la cadena. Devuelve un valor NULL si cualquier argumento de entrada es NULL.

Tipo de argumento:

- STRING, STRING
- Tipo de devolución INTEGER

Ejemplo:

```
os> source=people | eval `POSITION('world' IN 'helloworld')` = POSITION('world'
 IN 'helloworld'), `POSITION('invalid' IN 'helloworld')`= POSITION('invalid' IN
 'helloworld')  | fields `POSITION('world' IN 'helloworld')`, `POSITION('invalid' IN
 'helloworld')`
fetched rows / total rows = 1/1
+------------------------------------+---------------------------------------+
| POSITION('world' IN 'helloworld')   | POSITION('invalid' IN 'helloworld')   |
|------------------------------------+---------------------------------------|
| 6                                  | 0                                     |
+------------------------------------+---------------------------------------+
```

REVERSE

Uso: `REVERSE(str)` devuelve la cadena invertida de la cadena de entrada.

Tipo de argumento:

- STRING
- Tipo de devolución: STRING

Ejemplo:

```
os> source=people | eval `REVERSE('abcde')` = REVERSE('abcde') | fields
 `REVERSE('abcde')`
fetched rows / total rows = 1/1
+--------------------+
| REVERSE('abcde')   |
|--------------------|
| edcba              |
+--------------------+
```

RIGHT

Uso: `right(str, len)` devuelve los caracteres situados más a la derecha de la cadena de entrada. Devuelve 0 si no se encuentra la subcadena dentro de la cadena. Devuelve un valor NULL si cualquier argumento de entrada es NULL.

Tipo de argumento:

- STRING, INTEGER
- Tipo de devolución: STRING

Ejemplo:

```
os> source=people | eval `RIGHT('helloworld', 5)` = RIGHT('helloworld', 5),
 `RIGHT('HELLOWORLD', 0)` = RIGHT('HELLOWORLD', 0) | fields `RIGHT('helloworld', 5)`,
 `RIGHT('HELLOWORLD', 0)`
fetched rows / total rows = 1/1
+-------------------------+-------------------------+
| RIGHT('helloworld', 5)  | RIGHT('HELLOWORLD', 0)  |
|-------------------------+-------------------------|
| world                   |                         |
+-------------------------+-------------------------+
```

RTRIM

Uso: `rtrim(str)` recorta los espacios finales de la cadena de entrada.

Tipo de argumento:

- STRING
- Tipo de devolución: STRING

Ejemplo:

```
os> source=people | eval `RTRIM('   hello')` = RTRIM('   hello'), `RTRIM('hello   ')` =
 RTRIM('hello   ') | fields `RTRIM('   hello')`, `RTRIM('hello   ')`
fetched rows / total rows = 1/1
+--------------------+--------------------+
| RTRIM('   hello')  | RTRIM('hello   ')  |
|--------------------+--------------------|
|    hello           | hello              |
+--------------------+--------------------+
```

SUBSTRING

Uso: `substring(str, start)` o `substring(str, start, length)` devuelve una subcadena de la cadena de entrada. Sin especificar la longitud, devuelve la cadena completa desde la posición inicial.

Tipo de argumento:

- STRING, INTEGER, INTEGER
- Tipo de devolución: STRING

Ejemplo:

```
os> source=people | eval `SUBSTRING('helloworld', 5)` = SUBSTRING('helloworld',
 5), `SUBSTRING('helloworld', 5, 3)` = SUBSTRING('helloworld', 5, 3) | fields
 `SUBSTRING('helloworld', 5)`, `SUBSTRING('helloworld', 5, 3)`
fetched rows / total rows = 1/1
+------------------------------+---------------------------------+
| SUBSTRING('helloworld', 5)   | SUBSTRING('helloworld', 5, 3)   |
|------------------------------+---------------------------------|
| oworld                       | owo                             |
+------------------------------+---------------------------------+
```

TRIM

Uso: `trim(string)` elimina los espacios en blanco iniciales y finales de la cadena de entrada.

Tipo de argumento:

- STRING
- Tipo de devolución: STRING

Ejemplo:

```
os> source=people | eval `TRIM('   hello')` = TRIM('   hello'), `TRIM('hello   ')` =
 TRIM('hello   ') | fields `TRIM('   hello')`, `TRIM('hello   ')`
fetched rows / total rows = 1/1
+--------------------+--------------------+
| TRIM('   hello')   | TRIM('hello   ')   |
|--------------------+--------------------|
| hello              | hello              |
+--------------------+--------------------+
```

UPPER

Uso: `upper(string)` convierte la cadena de entrada en mayúsculas.

Tipo de argumento:

- STRING
- Tipo de devolución: STRING

Ejemplo:

```
os> source=people | eval `UPPER('helloworld')` = UPPER('helloworld'),
 `UPPER('HELLOWORLD')` = UPPER('HELLOWORLD') | fields `UPPER('helloworld')`,
 `UPPER('HELLOWORLD')`
fetched rows / total rows = 1/1
+----------------------+----------------------+
| UPPER('helloworld')   | UPPER('HELLOWORLD')   |
|----------------------+----------------------|
| HELLOWORLD            | HELLOWORLD            |
+----------------------+----------------------+
```

Función de conversión de tipos PPL

> **Note**
>
> Para ver qué integraciones de orígenes de datos de AWS admiten esta función PPL, consulte the section called “Funciones”.

`TRIM`

Uso: `cast(expr as dateType)` convierte el valor `expr` en `dataType` y devuelve el valor del `dataType`.

Se aplican las siguientes reglas de conversión:

Reglas de conversión de tipos

Src/Target	STRING	NUMBER	BOOLEANO	TIMESTAMP	DATE	TIME
STRING		Note1	Note1	TIMESTAMP()	FECHA()	TIME()
NUMBER	Note1		v!=0	N/A	N/A	N/A

Src/Target	STRING	NUMBER	BOOLEANO	TIMESTAMP	DATE	TIME
BOOLEANO	Note1	v?1:0		N/A	N/A	N/A
TIMESTAMP	Note1	N/A	N/A		FECHA()	TIME()
DATE	Note1	N/A	N/A	N/A		N/A
TIME	Note1	N/A	N/A	N/A	N/A	

Ejemplo de conversión a cadena:

```
os> source=people | eval `cbool` = CAST(true as string), `cint` = CAST(1 as string),
 `cdate` = CAST(CAST('2012-08-07' as date) as string) | fields `cbool`, `cint`, `cdate`
fetched rows / total rows = 1/1
+---------+--------+------------+
| cbool   | cint   | cdate      |
|---------+--------+------------|
| true    | 1      | 2012-08-07 |
+---------+--------+------------+
```

Ejemplo de conversión a número:

```
os> source=people | eval `cbool` = CAST(true as int), `cstring` = CAST('1' as int) |
 fields `cbool`, `cstring`
fetched rows / total rows = 1/1
+---------+-----------+
| cbool   | cstring   |
|---------+-----------|
| 1       | 1         |
+---------+-----------+
```

Ejemplo de conversión a fecha:

```
os> source=people | eval `cdate` = CAST('2012-08-07' as date), `ctime` =
 CAST('01:01:01' as time), `ctimestamp` = CAST('2012-08-07 01:01:01' as timestamp) |
 fields `cdate`, `ctime`, `ctimestamp`
fetched rows / total rows = 1/1
+------------+----------+---------------------+
| cdate      | ctime    | ctimestamp          |
|------------+----------+---------------------|
```

```
| 2012-08-07 | 01:01:01 | 2012-08-07 01:01:01 |
+------------+----------+---------------------+
```

Ejemplo de conversión a encadenado:

```
os> source=people | eval `cbool` = CAST(CAST(true as string) as boolean) | fields
 `cbool`
fetched rows / total rows = 1/1
+---------+
| cbool   |
|---------|
| True    |
+---------+
```

Supervisión de dominios Amazon OpenSearch Service

El monitoreo es una parte importante del mantenimiento de la fiabilidad, la disponibilidad y el rendimiento de Amazon OpenSearch Service y de las demás soluciones de AWS. AWS proporciona las siguientes herramientas para monitorear sus recursos de OpenSearch Service, informar de los problemas y tomar acciones automáticas cuando sea necesario:

Amazon CloudWatch

Amazon CloudWatch monitorea los recursos de OpenSearch Service en tiempo real. Puede recopilar métricas y realizar un seguimiento de ellas, crear paneles personalizados y definir alarmas que realizan advertencias o acciones cuando una métrica alcanza el umbral que se especifique. Para más información, consulte la Guía del usuario de Amazon CloudWatch.

Registros de Amazon CloudWatch

Amazon CloudWatch Logs permite monitorear, almacenar y acceder a sus archivos de registro de OpenSearch. CloudWatch Logs monitorea la información de los archivos de registro y puede notificarlo cuando se alcanzan ciertos umbrales. Para obtener más información, consulte la Guía del usuario de Registros de Amazon CloudWatch.

Amazon EventBridge

Amazon EventBridge ofrece un flujo casi en tiempo real de eventos del sistema que describen los cambios en sus dominios de OpenSearch Service. Puede crear reglas que vigilen determinados eventos y desencadenen acciones automatizadas en otros servicios de AWS cuando estos eventos se produzcan. Para más información, consulte la Guía del usuario de Amazon EventBridge.

AWS CloudTrail

AWS CloudTrail captura las llamadas de la API de configuración realizadas a OpenSearch Service como eventos. Puede entregar estos eventos al bucket de Amazon S3 que se especifique. Mediante esta información, puede identificar qué usuarios y cuentas realizaron solicitudes, la dirección IP fuente desde la que se realizaron las solicitudes y cuándo se produjeron. Para obtener más información, consulte la Guía del usuario de AWS CloudTrail.

Temas

- Monitoreo de métricas del clúster de OpenSearch con Amazon CloudWatch
- Monitoreo de registros de OpenSearch con Registros de Amazon CloudWatch

- Monitoreo de registros de auditoría en Amazon OpenSearch Service
- Monitoreo de eventos de OpenSearch Service con Amazon EventBridge
- Monitoreo de las llamadas a la API de Amazon OpenSearch Service con AWS CloudTrail

Monitoreo de métricas del clúster de OpenSearch con Amazon CloudWatch

Amazon OpenSearch Service publica datos de sus dominios en Amazon CloudWatch. CloudWatch permite recuperar las estadísticas sobre estos puntos de datos como un conjunto ordenado de datos de serie temporal denominado métricas. OpenSearch Service envía la mayoría de las métricas a CloudWatch en intervalos de 60 segundos. Si utiliza los volúmenes de EBS magnéticos o de uso general, las métricas de volumen de EBS solo se actualizan cada cinco minutos. Todas las métricas acumuladas (por ejemplo, `ThreadpoolWriteRejected`, `ThreadpoolSearchRejected`) están en la memoria y perderán estado. Las métricas se restablecerán durante la caída de un nodo, la devolución de un nodo, el reemplazo de un nodo y la implementación azul/verde. Para obtener más información sobre Amazon CloudWatch, consulte la Guía del usuario de Amazon CloudWatch.

La consola de OpenSearch Service muestra una serie de gráficos basados en los datos sin procesar de CloudWatch. En función de sus necesidades, es posible que prefiera ver los datos del clúster en CloudWatch en lugar de los gráficos que se muestran en la consola. El servicio archiva las métricas durante dos semanas antes de eliminarlas. Las métricas se proporcionan sin cargo adicional, pero CloudWatch sigue cobrando por la creación de paneles y alarmas. Para más información, consulte Precios de Amazon CloudWatch.

OpenSearch Service publica las siguientes métricas en CloudWatch:

- the section called “Métricas de clúster”
- the section called “Métricas de nodo maestro dedicado”
- the section called “Métricas de volumen de EBS”
- the section called “Métricas de la instancia”
- the section called “Métricas de UltraWarm”
- the section called “Métricas de nodos coordinadores dedicados”
- the section called “Métricas de almacenamiento en frío”
- the section called “Métricas de alertas”
- the section called “Métricas de detección de anomalías”

- the section called “Métricas de búsqueda asíncrona”
- the section called “Métricas de SQL”
- the section called “Métricas k-NN”
- the section called “Métricas de búsqueda entre clústeres”
- the section called “Métricas de replicación entre clústeres”
- the section called “Aprender a clasificar métricas”
- the section called “Métricas del lenguaje de procesamiento de canalizaciones”

Ver métricas con CloudWatch

Las métricas de CloudWatch se agrupan por el espacio de nombres de servicio y, a continuación, por las diversas combinaciones de dimensiones dentro de cada espacio de nombres.

Para consultar métricas mediante la consola de CloudWatch

1. Abra la consola de CloudWatch en https://console.aws.amazon.com/cloudwatch/.
2. En el panel de navegación, busque Métricas y elija, Todas las métricas. Seleccione el espacio de nombres ES/OpenSearchService.
3. Seleccione una dimensión para ver las métricas correspondientes. Las métricas de los nodos individuales se encuentran en la dimensión `ClientId, DomainName, NodeId`. Las métricas de clúster se encuentran en la dimensión `Per-Domain, Per-Client Metrics`. Algunas métricas de nodos se agregan a nivel de clúster y, por lo tanto, se incluyen en ambas dimensiones. Las métricas de particiones se encuentran en la dimensión `ClientId, DomainName, NodeId, ShardRole`.

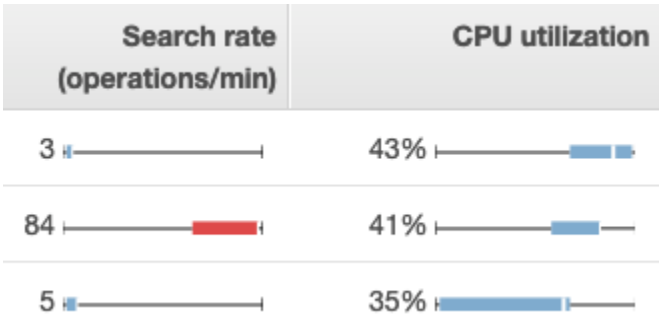
Visualización de una lista de métricas con la AWS CLI

Ejecute el siguiente comando:

```
aws cloudwatch list-metrics --namespace "AWS/ES"
```

Interpretación de gráficos de estado en OpenSearch Service

Para ver métricas en OpenSearch Service, utilice las pestañas Estado del clúster e Estado de la instancia. La pestaña Estado de la instancia utiliza gráficos de caja para proporcionar una visibilidad a simple vista del estado de cada nodo de OpenSearch:



- Cada cuadro de color muestra el rango de valores correspondiente al nodo durante el periodo de tiempo especificado.
- Los cuadros azules representan valores que son coherentes con otros nodos. Los cuadros rojos representan los valores atípicos.
- La línea blanca en cada cuadro muestra el valor actual del nodo.
- Los “bigotes” a cada lado del cuadro muestran los valores mínimo y máximo de todas las instancias a lo largo del periodo de tiempo.

Cuando se realizan cambios de configuración en el dominio, a menudo la lista de instancias individuales de las pestañas Estado del clúster e Estado de instancia duplica su tamaño durante un breve periodo antes de volver al número correcto. Para obtener una explicación de este comportamiento, consulte the section called “Cambios de configuración”.

Métricas de clúster

Amazon OpenSearch Service ofrece las siguientes métricas para clústeres.

Métrica	Descripción
`ClusterStatus.green`	Un valor de 1 indica que todas las particiones de índice se han asignado a los nodos del clúster. Estadísticas pertinentes: máximo
`ClusterStatus.yellow`	El valor 1 indica que las particiones principales de todos los índices se han asignado a nodos del clúster, pero las particiones de réplica de al menos un índice están sin asignar. Para más información, consulte the section called “Estado amarillo del clúster”.

Métrica	Descripción
	Estadísticas pertinentes: Máximo
`ClusterStatus.red`	Un valor de 1 indica que las particiones principal y replicada de al menos un índice no se han asignado a los nodos del clúster. Para más información, consulte the section called “Estado rojo del clúster”. Estadísticas pertinentes: máximo
`Shards.active`	El número total de particiones primarias y de réplicas activas. Estadísticas relevantes: Máximo, Suma
`Shards.unassigned`	El número de particiones que no se asignaron a los nodos del clúster. Estadísticas relevantes: Máximo, Suma
`Shards.delayedUnassigned`	El número de particiones cuya asignación de nodos se retrasó por la configuración de tiempo de espera. Estadísticas relevantes: Máximo, Suma
`Shards.activePrimary`	El número de particiones primarias activas. Estadísticas relevantes: Máximo, Suma
`Shards.initializing`	El número de particiones que se encuentran en inicialización. Estadísticas pertinentes: Suma
`Shards.relocating`	El número de particiones que se encuentran en reubicación. Estadísticas pertinentes: suma

Métrica	Descripción
`Nodes`	La cantidad de nodos en el clúster de OpenSearch Service, incluidos los nodos maestros dedicados y los nodos UltraWarm . Para más información, consulte the section called “Cambios de configuración”. Estadísticas pertinentes: máximo
`SearchableDocuments`	El número total de documentos que admiten búsquedas para todos los nodos de datos del clúster. Estadísticas pertinentes: Minimum, Maximum, Average
`DeletedDocuments`	El número total de documentos marcados para su eliminación en todos los nodos de datos del clúster. Estos documentos ya no aparecerán en los resultados de búsqueda, pero OpenSearch solo borra los documentos eliminados del disco durante las combinaciones de segmentos. Esta métrica aumenta después de las solicitudes de eliminación y disminuye después de las combinaciones de segmentos. Estadísticas pertinentes: Minimum, Maximum, Average
`CPUUtilization`	El porcentaje de uso de CPU para los nodos de datos del clúster. Máximo muestra el nodo con el mayor uso de CPU. Promedio representa todos los nodos del clúster. Esta métrica también está disponible para los nodos individuales. Estadísticas relevantes: máximo, promedio

Métrica	Descripción
`FreeStorageSpace`	El espacio libre para los nodos de datos del clúster. `Sum` muestra el espacio libre total del clúster, pero debe dejar el periodo en un minuto para recibir un valor preciso. `Minimum` y `Maximum` muestran los nodos con el mínimo y máximo espacio libre, respectivamente. Esta métrica también está disponible para los nodos individuales. OpenSearch Service muestra una `ClusterBlockException` cuando esta métrica alcanza `0`. Para efectuar la recuperación, debe eliminar índices, agregar instancias de mayor tamaño o bien agregar almacenamiento basado en EBS a las instancias existentes. Para más información, consulte the section called “Falta de espacio de almacenamiento disponible”. La consola de OpenSearch Service muestra este valor en GiB. La consola de Amazon CloudWatch lo muestra en MiB. **Note** `FreeStorageSpace` siempre será menor que los valores que el valor que proporcionan las API `_cluster/stats` y `_cat/allocation` de OpenSearch. OpenSearch Service reserva un porcentaje del espacio de almacenamiento de cada instancia para operaciones internas. Para más información, consulte Cálculo de requisitos de almacenamiento. Estadísticas pertinentes: mínimo, máximo, promedio, suma
`ClusterUsedSpace`	El espacio utilizado total para el clúster. Debe dejar el periodo en un minuto para obtener un valor preciso. La consola de OpenSearch Service muestra este valor en GiB. La consola de Amazon CloudWatch lo muestra en MiB. Estadísticas pertinentes: mínimo, máximo

Métrica	Descripción
`ClusterIndexWritesBlocked`	Indica si el clúster acepta o bloquea las solicitudes de escritura entrantes. Un valor de 0 indica que el clúster acepta solicitudes. Un valor de 1 indica que el clúster bloquea las solicitudes. Algunos factores comunes son los siguientes: `FreeStorageSpace` es demasiado bajo o `JVMMemoryPressure` demasiado alta. Para solucionar este problema, considere la posibilidad de agregar más espacio de disco o de ampliar el clúster. Estadísticas pertinentes: máximo
`JVMMemoryPressure`	El porcentaje máximo del montón de Java que se emplea para todos los nodos de datos del clúster. OpenSearch Service utiliza la mitad de la RAM de una instancia para la pila de Java, hasta un tamaño de 32 GiB. Puede escalar las instancias verticalmente hasta 64 GiB de RAM y después escalarlas horizontalmente mediante el agregado de instancias. Consulte the section called “Alarmas de CloudWatch recomendadas”. Estadísticas pertinentes: máximo **Note** La lógica de esta métrica cambió en el software del servicio R20220323. Para más información, consulte las notas de la versión.
`OldGenJVMMemoryPressure`	El porcentaje máximo de la pila de Java que se emplea para la “anterior generación” en todos los nodos de datos del clúster. Esta métrica también está disponible a nivel de nodo. Estadísticas pertinentes: máximo

Métrica	Descripción
AutomatedSnapshotFailure	El número de instantáneas automatizadas que han producido un error para el clúster. Un valor de 1 indica que no se ha tomado ninguna instantánea automatizada del dominio en las últimas 36 horas. Estadísticas pertinentes: mínimo, máximo
CPUCreditBalance	Los créditos de CPU restantes que están disponibles para los nodos de datos del clúster. Un crédito de CPU proporciona el rendimiento de un núcleo de CPU completo durante un minuto. Para más información, consulte Créditos CPU en la Guía para desarrolladores de Amazon EC2. Esta métrica solo está disponible para el tipo de instancias T2. Estadísticas pertinentes: mínimo
OpenSearchDashboardsHealthyNodes	Una comprobación de estado de OpenSearch Dashboards. Si el mínimo, el máximo y el promedio son todos iguales a 1, Dashboards se comporta con normalidad. Si tiene 10 nodos con un máximo de 1, mínimo de 0 y promedio de 0,7, esto significa que 7 nodos (70 %) están en buen estado y 3 nodos (30 %) no lo están. Estadísticas pertinentes: Minimum, Maximum, Average
OpensearchDashboardsReportingFailedRequestSysErrCount	El número de solicitudes para generar informes de OpenSearch Dashboards que tuvieron errores debido a problemas del servidor o a limitaciones de las características. Estadísticas pertinentes: suma
OpensearchDashboardsReportingFailedRequestUserErrCount	El número de solicitudes para generar informes de OpenSearch Dashboards que no se realizaron de forma correcta debido a problemas del cliente. Estadísticas pertinentes: suma

Métrica	Descripción
OpensearchDashboardsReportingRequestCount	El número total de solicitudes para generar informes de OpenSearch Dashboards. Estadísticas pertinentes: suma
OpensearchDashboardsReportingSuccessCount	El número de solicitudes realizadas de forma correcta para generar informes de OpenSearch Dashboards. Estadísticas pertinentes: suma
KMSKeyError	Un valor igual a 1 indica que la clave de AWS KMS utilizada para cifrar los datos en reposo se deshabilitó. Para restablecer el dominio a operaciones normales, rehabilita la clave. La consola solo muestra esta métrica para los dominios que encriptan los datos en reposo. Estadísticas pertinentes: mínimo, máximo
KMSKeyInaccessible	Un valor igual a 1 indica que la clave de AWS KMS utilizada para cifrar los datos en reposo se eliminó o se revocaron sus permisos a OpenSearch Service. No puede recuperar los dominios que están en este estado. Sin embargo, si tiene una instantánea manual, puede utilizarla para migrar los datos del dominio a un nuevo dominio. La consola solo muestra esta métrica para los dominios que encriptan los datos en reposo. Estadísticas pertinentes: mínimo, máximo

Métrica	Descripción
`InvalidHostHeaderRequests`	El número de solicitudes HTTP realizadas al clúster de OpenSearch que incluía un encabezado de host no válido (o faltante). Las solicitudes válidas incluyen el nombre de host del dominio como valor de encabezado de host. OpenSearch Service rechaza las solicitudes no válidas para dominios de acceso público que no tienen una política de acceso restrictiva. Es recomendable que aplique una política de acceso restrictiva a todos los dominios. Si se muestran valores altos para esta métrica, confirme que sus clientes de OpenSearch incluyen el nombre de host del dominio (y no, por ejemplo, su dirección IP) en sus solicitudes. Estadísticas pertinentes: suma
`OpenSearchRequests (previously ElasticsearchRequests)`	El número de solicitudes realizadas al clúster de OpenSearch. Estadísticas pertinentes: suma
`2xx, 3xx, 4xx, 5xx`	El número de solicitudes para el dominio que produjeron el código de respuesta HTTP especificado (2xx, 3xx, 4xx, 5xx). Estadísticas pertinentes: suma

Métrica	Descripción
`ThroughputThrottle`	Indica si los discos se han limitado o no. La limitación se produce cuando el rendimiento combinado de `ReadThroughputMicroBursting` y `WriteThroughputMicroBursting` es superior al rendimiento máximo, `MaxProvisionedThroughput`. `MaxProvisionedThroughput` es el valor más bajo del rendimiento de la instancia o del volumen aprovisionado. Un valor de 1 indica que los discos se han limitado. Un valor de 0 indica un comportamiento normal. Para obtener más información sobre el rendimiento de la instancia, consulte Instancias optimizadas para Amazon EBS. Para obtener información sobre el rendimiento del volumen, consulte Tipos de volúmenes de Amazon EBS. Estadísticas pertinentes: mínimo, máximo
`IopsThrottle`	Indica si se ha limitado o no el número de operaciones de entrada/salida por segundo (IOPS) en el dominio. La limitación se produce cuando las IOPS del nodo de datos superan el límite máximo permitido del volumen de EBS o de la instancia de EC2 del nodo de datos. Para obtener más información sobre IOPS de instancias, consulte Tipos de instancias optimizadas para Amazon EBS. Para obtener información sobre IOPS de volúmenes, consulte Amazon EBS volume types. Estadísticas pertinentes: mínimo, máximo
`HighSwapUsage`	Un valor de 1 indica que el intercambio debido a errores de página puede provocar picos en el uso del disco subyacente durante un periodo de tiempo específico. Estadísticas pertinentes: máximo

Métricas de nodo maestro dedicado

Amazon OpenSearch Service ofrece las siguientes métricas para nodos maestros dedicados.

Métrica	Descripción
`MasterCPUUtilization`	El porcentaje máximo de recursos de CPU que se utilizan en los nodos maestros dedicados. Recomendamos aumentar el tamaño del tipo de instancia cuando esta métrica alcance el 60 por ciento. Estadísticas pertinentes: máximo
`MasterFreeStorageSpace`	Esta métrica no es pertinente y puede pasarse por alto. El servicio no usa los nodos maestros como nodos de datos.
`MasterJVMMemoryPressure`	El porcentaje máximo del montón de Java que se emplea para todos los nodos maestros dedicados del clúster. Recomendamos cambiar a un tipo de instancia mayor cuando esta métrica alcance el 85 por ciento. Estadísticas pertinentes: máximo **Note** La lógica de esta métrica cambió en el software del servicio R20220323. Para más información, consulte las notas de la versión.
`MasterOldGenJVMMemoryPressure`	El porcentaje máximo de la pila de Java que se emplea para la “anterior generación” según el nodo maestro. Estadísticas pertinentes: máximo
`MasterCPUCreditBalance`	Los créditos de CPU restantes que están disponibles para los nodos maestros dedicados del clúster. Un crédito de CPU proporciona el rendimiento de un núcleo de CPU completo durante un minuto. Para más información, consulte Créditos CPU en la Guía para desarrolladores de Amazon EC2. Esta métrica solo está disponible para el tipo de instancias T2.

Métrica	Descripción
	Estadísticas pertinentes: mínimo
`MasterReachableFromNode`	Comprobación de estado de las excepciones `MasterNotDiscovered` . Un valor de 1 indica un comportamiento normal. Un valor de 0 indica que `/_cluster/health/` ha dado error. Los errores significan que no se puede acceder al nodo maestro desde el nodo de origen. Suelen ser el resultado de un problema de conectividad de red o de dependencia de AWS. Estadísticas pertinentes: máximo
`MasterSysMemoryUtilization`	El porcentaje de memoria del nodo maestro que está en uso. Estadísticas pertinentes: máximo

Métricas de nodos coordinadores dedicados

Amazon OpenSearch Service ofrece las siguientes métricas para nodos coordinadores dedicados.

Métrica	Descripción
`CoordinatorCPUUtilization`	El porcentaje máximo de recursos de CPU que se utilizan en los nodos coordinadores dedicados. Recomendamos aumentar el tamaño del tipo de instancia cuando esta métrica alcance el 80 %. Estadísticas pertinentes: máximo
`CoordinatorJVMMemoryPressure`	El porcentaje máximo del montón de Java que se emplea para todos los nodos coordinadores dedicados del clúster. Recomendamos cambiar a un tipo de instancia mayor cuando esta métrica alcance el 85 por ciento. Estadísticas pertinentes: máximo
`CoordinatorOldGenJVMMemoryPressure`	El porcentaje máximo de la pila de Java que se emplea para la “anterior generación” según el nodo maestro.

Métrica	Descripción
	Estadísticas pertinentes: máximo
`CoordinatorSysMemoryUtilization`	El porcentaje de la memoria del nodo coordinador que está en uso. Estadísticas pertinentes: máximo
`CoordinatorFreeStorageSpace`	Esta métrica indica que el servicio no usa los nodos coordinadores como nodos de datos.

Métricas de volumen de EBS

Amazon OpenSearch Service ofrece las siguientes métricas para volúmenes de EBS.

Métrica	Descripción
`ReadLatency`	La latencia en segundos para las operaciones de lectura en los volúmenes de EBS. Esta métrica también está disponible para los nodos individuales. Estadísticas pertinentes: Minimum, Maximum, Average
`WriteLatency`	La latencia en segundos para las operaciones de escritura en los volúmenes de EBS. Esta métrica también está disponible para los nodos individuales. Estadísticas pertinentes: Minimum, Maximum, Average
`ReadThroughput`	El rendimiento en bytes por segundo para las operaciones de lectura en los volúmenes de EBS. Esta métrica también está disponible para los nodos individuales. Estadísticas pertinentes: Minimum, Maximum, Average
`ReadThroughputMicroBursting`	Se tiene en cuenta el rendimiento, en bytes por segundo, de las operaciones de lectura en volúmenes de EBS cuando se tiene en cuenta la microrráfaga. Esta métrica también está disponible para los nodos individuales. La microrráfaga se produce cuando un volumen de EBS

Métrica	Descripción
	alcanza niveles altos de IOPS o rendimiento durante períodos de tiempo significativamente más cortos (menos de un minuto). Estadísticas pertinentes: Minimum, Maximum, Average
`WriteThroughput`	El rendimiento en bytes por segundo para las operaciones de escritura en los volúmenes de EBS. Esta métrica también está disponible para los nodos individuales. Estadísticas pertinentes: Minimum, Maximum, Average
`WriteThroughputMicroBursting`	Se tiene en cuenta el rendimiento, en bytes por segundo, de las operaciones de escritura en volúmenes de EBS cuando se tiene en cuenta la microrráfaga. Esta métrica también está disponible para los nodos individuales. La microrráfaga se produce cuando un volumen de EBS alcanza niveles altos de IOPS o rendimiento durante períodos de tiempo significativamente más cortos (menos de un minuto). Estadísticas pertinentes: Minimum, Maximum, Average
`DiskQueueDepth`	El número de solicitudes de entrada y salida (E/S) pendientes de un volumen de EBS. Estadísticas pertinentes: Minimum, Maximum, Average
`ReadIOPS`	El número de operaciones de entrada y salida (E/S) por segundo para las operaciones de lectura en los volúmenes de EBS. Esta métrica también está disponible para los nodos individuales. Estadísticas pertinentes: Minimum, Maximum, Average

Métrica	Descripción
`ReadIOPSMicroBursting`	El número de operaciones de entrada y salida (E/S) por segundo para las operaciones de lectura en los volúmenes de EBS cuando se tiene en cuenta la microrráfaga. Esta métrica también está disponible para los nodos individuales. La microrráfaga se produce cuando un volumen de EBS alcanza niveles altos de IOPS o rendimiento durante períodos de tiempo significativamente más cortos (menos de un minuto). Estadísticas pertinentes: Minimum, Maximum, Average
`WriteIOPS`	El número de operaciones de entrada y salida (E/S) por segundo para las operaciones de escritura en los volúmenes de EBS. Esta métrica también está disponible para los nodos individuales. Estadísticas pertinentes: Minimum, Maximum, Average
`WriteIOPS MicroBursting`	El número de operaciones de entrada y salida (E/S) por segundo para las operaciones de escritura en los volúmenes de EBS cuando se tiene en cuenta la microrráfaga. Esta métrica también está disponible para los nodos individuales. La microrráfaga se produce cuando un volumen de EBS alcanza niveles altos de IOPS o rendimiento durante períodos de tiempo significativamente más cortos (menos de un minuto). Estadísticas pertinentes: Minimum, Maximum, Average
`BurstBalance`	El porcentaje de los créditos de entrada y salida (E/S) que quedan en el bucket de ráfaga para un volumen de EBS. Un valor de 100 significa que el volumen ha acumulado el número máximo de créditos. Si este porcentaje cae por debajo del 70 %, consulte the section called “Bajo balance de ráfaga EBS”. El balance de ráfagas se mantiene en 0 para los dominios con tipos de volúmenes gp3, así como para los dominios con volúmenes gp2 que tengan un tamaño de volumen superior a 1000 GiB. Estadísticas pertinentes: Minimum, Maximum, Average

Métrica	Descripción
`VolumeStalledIOcheck`	El estado de sus volúmenes EBS para determinar cuándo están deteriorados. La métrica es un valor binario que devuelve un estado 0 (aprobado) o 1 (fallido) según si el volumen EBS puede completar las operaciones de entrada y salida. `VolumeStalledIOcheck` también está disponible para nodos individuales. Estadísticas pertinentes: Minimum, Maximum, Average

Métricas de la instancia

Amazon OpenSearch Service proporciona las siguientes métricas para cada instancia de un dominio. OpenSearch Service también acumula estas métricas de instancia para proporcionar información sobre el estado general del clúster. Puede verificar este comportamiento mediante la estadística Recuento de muestras en la consola. Tenga en cuenta que las métricas de la tabla siguiente disponen de estadísticas relevantes para el nodo y el clúster.

> **Important**
>
> Las diferentes versiones de Elasticsearch utilizan diferentes grupos de subprocesos para procesar las llamadas a la API `_index`. Elasticsearch 1.5 y 2.3 utilizan el grupo de subprocesos de indexación. Elasticsearch 5.x, 6.0 y 6.2 utilizan el grupo de subprocesos por lotes. OpenSearch y Elasticsearch 6.3 y posteriores utilizan el grupo de subprocesos de escritura. Actualmente, la consola de OpenSearch Service no incluye un gráfico para el grupo de subprocesos masivos.
> Utilice `GET _cluster/settings?include_defaults=true` para verificar el tamaño del grupo de subprocesos y las colas para el clúster.

Métrica	Descripción
`FetchLatency`	La diferencia en el tiempo total, en milisegundos, que tardan todas las operaciones de obtención de particiones en un nodo entre el minuto N y el minuto (N - 1). Estadísticas de nodo pertinentes: promedio

Métrica	Descripción
	Estadísticas de clúster pertinentes: promedio, máximo
`FetchRate`	El número total de operaciones de obtención de particiones por minuto para todos los particiones en un nodo de datos. Estadísticas de nodo pertinentes: promedio Estadísticas de clúster pertinentes: promedio, máximo, suma
`ScrollTotal`	El número total de operaciones de desplazamiento de particiones por minuto para todas las particiones en un nodo de datos. Estadísticas de nodo relevantes: Promedio, Máximo Estadísticas de clúster pertinentes: promedio, máximo, suma
`ScrollCurrent`	El número de operaciones de desplazamiento de particiones que se están ejecutando actualmente. Estadísticas de nodo relevantes: Promedio, Máximo Estadísticas de clúster pertinentes: promedio, máximo, suma
`OpenContexts`	El número de contextos de búsqueda abiertos. Estadísticas de nodo relevantes: Promedio, Máximo Estadísticas de clúster pertinentes: promedio, máximo, suma
`ThreadCount`	El número total de hilos que actualmente está utilizando el proceso de OpenSearch. Estadísticas de nodo relevantes: Promedio, Máximo Estadísticas de clúster pertinentes: promedio, máximo, suma

Métrica	Descripción
`ShardReactivateCount`	El número total de veces que todos los particiones se han activado desde un estado inactivo. Estadísticas de nodo relevantes: Suma, Máximo Estadísticas de clúster relevantes: Suma, Máximo
`ConcurrentSearchRate`	El número total de solicitudes de búsqueda con búsqueda de segmentos simultánea por minuto de todas las particiones de un nodo de datos. Una sola llamada a la API `_search` podría devolver resultados de muchas particiones diferentes. Si cinco de estas particiones se encuentran en un nodo, este notificaría el valor 5 para la métrica, incluso aunque el cliente solo realizara una solicitud. Estadísticas de nodo pertinentes: promedio Estadísticas de clúster pertinentes: promedio, máximo, suma
`ConcurrentSearchLatency`	La diferencia en el tiempo total, en milisegundos, calculada por todas las búsquedas que usan la búsqueda de segmentos simultánea en un nodo entre el minuto N y el minuto (N-1). Estadísticas de nodo pertinentes: promedio Estadísticas de clúster pertinentes: promedio, máximo
`IndexingLatency`	La diferencia en el tiempo total, en milisegundos, calculada por todas las operaciones de indexación en un nodo entre el minuto N y el minuto (N-1). Estadísticas de nodo pertinentes: promedio Estadísticas de clúster pertinentes: promedio, máximo

Métrica	Descripción
`IndexingRate`	El número de operaciones de indexación por minuto. Una sola llamada a la API `_bulk` que agrega dos documentos y actualiza dos recuentos como cuatro operaciones, que podrían propagars e por uno o varios nodos. Si dicho índice tiene una o varias réplicas y está en un dominio de OpenSearch sin instancias optimizadas, otros nodos del clúster también registran un total de cuatro operaciones de indexación. En el caso de los dominios de OpenSearch con instancias optimizadas, otros nodos con réplicas no registran ninguna operación. Las eliminaciones de documentos no se tienen en cuenta en esta métrica. Estadísticas de nodo pertinentes: promedio Estadísticas de clúster pertinentes: promedio, máximo, suma
`SearchLatency`	La diferencia en el tiempo total, en milisegundos, calculada por todas las búsquedas en un nodo entre el minuto N y el minuto (N-1). Estadísticas de nodo pertinentes: promedio Estadísticas de clúster pertinentes: promedio, máximo
`SearchRate`	El número total de peticiones de búsqueda por minuto de todas las particiones de un nodo de datos. Una sola llamada a la API `_search` podría devolver resultados de muchas particiones diferentes. Si cinco de estas particiones se encuentran en un nodo, este notificaría el valor 5 para la métrica, incluso aunque el cliente solo realizara una solicitud. Estadísticas de nodo pertinentes: promedio Estadísticas de clúster pertinentes: promedio, máximo, suma

Métrica	Descripción
SegmentCount	El número de segmentos de un nodo de datos. Cuantos más segmentos tenga, más tiempo tardará cada búsqueda. OpenSearch de vez en cuando combina segmentos más pequeños en uno más grande. Estadísticas de nodo pertinentes: máximo, promedio Estadísticas de clúster pertinentes: suma, máximo, promedio
SysMemoryUtilization	El porcentaje de memoria de la instancia que está en uso. Es normal obtener valores altos para esta métrica y, por lo general, no representan un problema con el clúster. Para obtener un mejor indicador de posibles problemas de rendimiento y estabilidad, consulte la métrica JVMMemoryPressure . Estadísticas de nodo pertinentes: mínimo, máximo, promedio Estadísticas de clúster pertinentes: mínimo, máximo, promedio
JVMGCYoungCollectionCount	Número de veces que se ha ejecutado la recopilación de elementos no utilizados “nueva generación”. Un número grande y en continuo crecimiento de ejecuciones es normal para las operaciones de clúster. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: suma, máximo, promedio
JVMGCYoungCollectionTime	La cantidad de tiempo en milisegundos que el clúster ha empleado en realizar la recopilación de elementos no utilizados “nueva generación”. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: suma, máximo, promedio

Métrica	Descripción
`JVMGCOldCollectionCount`	Número de veces que se ha ejecutado la recopilación de elementos no utilizados “generación anterior”. En un clúster con suficientes recursos, este número debe ser pequeño y crecer con poca frecuencia. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: suma, máximo, promedio
`JVMGCOldCollectionTime`	La cantidad de tiempo en milisegundos que el clúster ha empleado en realizar la recopilación de elementos no utilizados “generación anterior”. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: suma, máximo, promedio
`OpenSearchDashboardsConcurrentConnections`	El número de conexiones simultáneas activas a OpenSearch Dashboards. Si este número crece continuamente, considere la posibilidad de escalar el clúster. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: suma, máximo, promedio
`OpenSearchDashboardsHealthyNode`	Comprobación de estado del nodo individual de OpenSearch Dashboards. Un valor de 1 indica un comportamiento normal. Un valor de 0 indica que Dashboards es inaccesible. Estadísticas de nodo pertinentes: mínimo Estadísticas de clúster pertinentes: mínimo, máximo, promedio

Métrica	Descripción
OpenSearchDashboardsHeapTotal	La cantidad de memoria de pila asignada a OpenSearch Dashboards en MiB. Los diferentes tipos de instancias EC2 pueden afectar a la asignación exacta de memoria. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: suma, máximo, promedio
OpenSearchDashboardsHeapUsed	La cantidad absoluta de memoria de la pila utilizada por OpenSearch Dashboards en MiB. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: suma, máximo, promedio
OpenSearchDashboardsHeapUtilization	El porcentaje máximo de memoria de la pila disponible que utiliza OpenSearch Dashboards. Si este valor aumenta por encima del 80 %, considere la posibilidad de escalar el clúster. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: mínimo, máximo, promedio
OpenSearchDashboardsOS1MinuteLoad	El promedio de carga de CPU de un minuto para OpenSearch Dashboards. La carga de la CPU idealmente debería permanecer por debajo de 1,00. Aunque los picos temporales están bien, se recomienda aumentar el tamaño del tipo de instancias si esta métrica está constantemente por encima de 1,00. Estadísticas de nodo pertinentes: promedio Estadísticas de clúster pertinentes: promedio, máximo

Métrica	Descripción
`OpenSearchDashboardsRequestTotal`	El recuento total de solicitudes HTTP realizadas a OpenSearch Dashboards. Si su sistema es lento o ve un número elevado de solicitudes de Dashboards, considere aumentar el tamaño del tipo de instancias. Estadísticas de nodos pertinentes: suma Estadísticas de clúster pertinentes: suma
`OpenSearchDashboardsResponseTimesMaxInMillis`	El tiempo máximo, en milisegundos, que OpenSearch Dashboards tarda en responder a una solicitud. Si las solicitudes tardan mucho tiempo en devolver resultados, considere aumentar el tamaño del tipo de instancias. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: máximo, promedio
`SearchTaskCancelled`	El número de cancelaciones del nodo coordinador. Estadísticas de nodos pertinentes: suma Estadísticas de clúster pertinentes: suma
`SearchShardTaskCancelled`	El número de cancelaciones del nodo de datos. Estadísticas de nodos pertinentes: suma Estadísticas de clúster pertinentes: suma,
`ThreadpoolForce_mergeQueue`	El número de tareas en cola en el grupo de subprocesos de combinación forzada. Si el tamaño de la cola es sistemáticamente elevado, considere la posibilidad de escalar el clúster. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: suma, máximo, promedio

Métrica	Descripción
ThreadpoolForce_mergeRejected	El número de tareas rechazadas en el grupo de subprocesos de combinación forzada. Si este número crece continuamente, considere la posibilidad de escalar el clúster. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: suma
ThreadpoolForce_mergeThreads	El tamaño del grupo de subprocesos de combinación forzada. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: promedio, suma
ThreadpoolIndexQueue	El número de tareas en cola en el grupo de subprocesos de indexación. Si el tamaño de la cola es sistemáticamente elevado, considere la posibilidad de escalar el clúster. El tamaño de la cola de indexación máximo es 200. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: suma, máximo, promedio
ThreadpoolIndexRejected	El número de tareas rechazadas en el grupo de subprocesos de indexación. Si este número crece continuamente, considere la posibilidad de escalar el clúster. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: suma
ThreadpoolIndexThreads	El tamaño del grupo de subprocesos de indexación. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: promedio, suma

Métrica	Descripción
ThreadpoolSearchQueue	El número de tareas en cola en el grupo de subprocesos de búsqueda. Si el tamaño de la cola es sistemáticamente elevado, considere la posibilidad de escalar el clúster. El tamaño máximo de la cola de búsqueda es 1000. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: suma, máximo, promedio
ThreadpoolSearchRejected	El número de tareas rechazadas en el grupo de subprocesos de búsqueda. Si este número crece continuamente, considere la posibilidad de escalar el clúster. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: suma
ThreadpoolSearchThreads	El tamaño del grupo de subprocesos de búsqueda. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: promedio, suma
Threadpoolsql-workerQueue	El número de tareas en cola en el grupo de subprocesos de búsqueda de SQL. Si el tamaño de la cola es sistemáticamente elevado, considere la posibilidad de escalar el clúster. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: suma, máximo, promedio
Threadpoolsql-workerRejected	El número de tareas rechazadas en el grupo de subprocesos de búsqueda de SQL. Si este número crece continuamente, considere la posibilidad de escalar el clúster. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: suma

Métrica	Descripción
`Threadpoolsql-workerThreads`	El tamaño del grupo de subprocesos de búsqueda de SQL. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: promedio, suma
`ThreadpoolBulkQueue`	El número de tareas en cola en el grupo de subprocesos por lotes. Si el tamaño de la cola es sistemáticamente elevado, considere la posibilidad de escalar el clúster. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: suma, máximo, promedio
`ThreadpoolBulkRejected`	El número de tareas rechazadas en el grupo de subprocesos por lotes. Si este número crece continuamente, considere la posibilidad de escalar el clúster. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: suma
`ThreadpoolBulkThreads`	El tamaño del grupo de subprocesos por lotes. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: promedio, suma
`ThreadpoolIndexSearcherQueue`	El número de tareas en cola del grupo de subprocesos del buscador de índices. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: suma, máximo, promedio

Métrica	Descripción
ThreadpoolIndexSearcherRejected	El número de tareas rechazadas en el grupo de subprocesos del buscador de índices. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: suma
ThreadpoolIndexSearcherThreads	El tamaño del grupo de subprocesos del buscador de índices. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: promedio, suma
ThreadpoolWriteThreads	El tamaño del grupo de subprocesos de escritura. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: promedio, suma
ThreadpoolWriteQueue	El número de tareas en cola en el grupo de subprocesos de escritura. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: promedio, suma

Métrica	Descripción
`ThreadpoolWriteRejected`	El número de tareas rechazadas en el grupo de subprocesos de escritura. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: promedio, suma **Note** Debido a que el tamaño predeterminado de la cola de escritura aumentó de 200 a 10 000 en la versión 7.1, esta métrica ya no es el único indicador de rechazos de OpenSearch Service. Utilice las métricas `CoordinatingWriteRejected` , `PrimaryWriteRejected` y `ReplicaWriteRejected` para monitorear los rechazos en las versiones 7.1 y posteriores.
`CoordinatingWriteRejected`	El número total de rechazos se produjo en el nodo de coordinación debido a la presión de indexación desde el último inicio del proceso de OpenSearch Service. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: promedio, suma Esta métrica está disponible en la versión 7.1 y posteriores.
`PrimaryWriteRejected`	El número total de rechazos se produjo en las particiones principales debido a la presión de indexación desde el último inicio del proceso de OpenSearch Service. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: promedio, suma Esta métrica está disponible en la versión 7.1 y posteriores.

Métrica	Descripción
`ReplicaWriteRejected`	El número total de rechazos se produjo en las particiones de réplica debido a la presión de indexación desde el último inicio del proceso de OpenSearch Service. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: promedio, suma Esta métrica está disponible en la versión 7.1 y posteriores.
`WorkloadManagementEnabled`	Indica si la característica de administración de cargas de trabajo está habilitada. Un valor de 1 significa que está habilitada y un valor de 0 significa que está `monitor_only` o deshabilitada. Estadísticas de nodo relevantes: Máximo, Mínimo Estadísticas de clúster pertinentes: promedio, suma Esta métrica está disponible en la versión 7.1 y posteriores.
`SoftQueryGroupCount`	Número de grupos de consultas en modo flexible en el dominio. Estadísticas de nodo relevantes: Promedio, Máximo Estadísticas de clúster pertinentes: promedio, máximo, suma Esta métrica está disponible en la versión 7.1 y posteriores.
`EnforcedQueryGroupCount`	Número de grupos de consultas en modo aplicado en el dominio. Estadísticas de nodo relevantes: Promedio, Máximo Estadísticas de clúster pertinentes: promedio, máximo, suma Esta métrica está disponible en la versión 7.1 y posteriores.

Métricas de UltraWarm

Amazon OpenSearch Service ofrece las siguientes métricas para nodos de UltraWarm.

Métrica	Descripción
`WarmCPUUtilization`	El porcentaje de uso de CPU para los nodos UltraWarm del clúster. Máximo muestra el nodo con el mayor uso de CPU. Promedio represent a todos los nodos UltraWarm del clúster. Esta métrica también está disponible para nodos UltraWarm individuales. Estadísticas relevantes: máximo, promedio
`WarmFreeStorageSpace`	La cantidad de espacio de almacenamiento en caliente libre en MiB. Debido a que UltraWarm utiliza Amazon S3 en lugar de discos adjuntos, `Sum` es la única estadística pertinente. Debe dejar el periodo en un minuto para obtener un valor preciso. Estadísticas pertinentes: suma
`WarmSearchableDocuments`	Número total de documentos en los que se pueden realizar búsquedas de todos los índices templados del clúster. Debe dejar el periodo en un minuto para obtener un valor preciso. Estadísticas pertinentes: suma
`WarmSearchLatency`	La diferencia en el tiempo total, en milisegundos, calculada por todas las búsquedas en UltraWarm entre el minuto N y el minuto (N-1). Estadísticas de nodo pertinentes: promedio Estadísticas de clúster pertinentes: promedio, máximo
`WarmSearchRate`	El número total de peticiones de búsqueda por minuto de todas las particiones de un nodo UltraWarm. Una sola llamada a la API `_search` podría devolver resultados de muchas particiones diferentes. Si cinco de estas particiones se encuentran en un nodo, este notificaría el valor 5 para la métrica, incluso aunque el cliente solo realizara una solicitud. Estadísticas de nodo pertinentes: promedio Estadísticas de clúster pertinentes: promedio, máximo, suma

Métrica	Descripción
`WarmStorageSpaceUtilization`	Cantidad total del espacio de almacenamiento templado, en MiB, que utiliza el clúster. Estadísticas pertinentes: máximo
`HotStorageSpaceUtilization`	Cantidad total del espacio de almacenamiento en caliente que utiliza el clúster. Estadísticas pertinentes: máximo
`WarmSysMemoryUtilization`	Porcentaje de la memoria del nodo maestro que está en uso. Estadísticas pertinentes: máximo
`HotToWarmMigrationQueueSize`	Número de índices a la espera actualmente para migrar del almacenamiento caliente al almacenamiento templado. Estadísticas pertinentes: máximo
`WarmToHotMigrationQueueSize`	Número de índices a la espera actualmente para migrar del almacenamiento templado al almacenamiento caliente. Estadísticas pertinentes: máximo
`HotToWarmMigrationFailureCount`	El número total de errores de migración del almacenamiento en caliente al almacenamiento templado. Estadísticas pertinentes: suma
`HotToWarmMigrationForceMergeLatency`	Latencia media de la etapa de fusión de fuerzas del proceso de migración. Si esta etapa lleva demasiado tiempo, considere aumentar `index.ultrawarm.migration.force_merge.max_num_segments` . Estadísticas pertinentes: promedio

Métrica	Descripción
`HotToWarmMigrationSnapshotLatency`	Latencia media de la etapa de instantánea del proceso de migración. Si esta etapa tarda demasiado tiempo, asegúrese de que las particiones estén correctamente dimensionadas y distribuidas por todo el clúster. Estadísticas pertinentes: promedio
`HotToWarmMigrationProcessingLatency`	La latencia media de migraciones exitosas del almacenamiento en caliente al almacenamiento templado, sin incluir el tiempo que estuvo en la cola. Este valor es la suma de la cantidad de tiempo que se tarda en completar las etapas de fusión de fuerza, instantánea y reubicación de particiones del proceso de migración. Estadísticas pertinentes: promedio
`HotToWarmMigrationSuccessCount`	El número total de migraciones exitosas del almacenamiento en caliente al almacenamiento templado. Estadísticas pertinentes: suma
`HotToWarmMigrationSuccessLatency`	La latencia media de migraciones exitosas del almacenamiento en caliente al almacenamiento templado, incluido el tiempo que estuvo en la cola. Estadísticas pertinentes: promedio
`WarmThreadpoolSearchThreads`	El tamaño del grupo de subprocesos de búsqueda de UltraWarm. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: promedio, suma
`WarmThreadpoolSearchRejected`	El número de tareas rechazadas en el grupo de subprocesos de búsqueda de UltraWarm. Si este número crece, considere la posibilidad de agregar más nodos de UltraWarm. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: suma

Métrica	Descripción
`WarmThreadpoolSearchQueue`	El número de tareas en cola en el grupo de subprocesos de búsqueda de UltraWarm. Si el tamaño de la cola es elevado, considere la posibilidad de agregar más nodos de UltraWarm. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: suma, máximo, promedio
`WarmJVMMemoryPressure`	Porcentaje máximo del montón de Java que se emplea con los nodos de UltraWarm. Estadísticas pertinentes: máximo **Note** La lógica de esta métrica cambió en el software del servicio R20220323. Para más información, consulte las notas de la versión.
`WarmOldGenJVMMemoryPressure`	El porcentaje máximo de la pila de Java que se emplea para la “anterior generación” según el nodo UltraWarm. Estadísticas pertinentes: máximo
`WarmJVMGCYoungCollectionCount`	Número de veces que se ejecutó la recolección de basura de la “nueva generación” en los nodos de UltraWarm. Un número grande y en continuo crecimiento de ejecuciones es normal para las operaciones de clúster. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: suma, máximo, promedio

Métrica	Descripción
`WarmJVMGCYoungCollectionTime`	La cantidad de tiempo en milisegundos que el clúster empleó en realizar la recolección de basura de la “nueva generación” en los nodos de UltraWarm. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: suma, máximo, promedio
`WarmJVMGCOldCollectionCount`	Número de veces que se ejecutó la recolección de basura de la “generación anterior” en los nodos de UltraWarm. En un clúster con suficientes recursos, este número debe ser pequeño y crecer con poca frecuencia. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: suma, máximo, promedio
`WarmConcurrentSearchRate`	El número total de solicitudes de búsqueda con búsqueda de segmentos simultánea por minuto de todas las particiones de un nodo UltraWarm . Una sola llamada a la API `_search` podría devolver resultados de muchas particiones diferentes. Si cinco de estas particiones se encuentran en un nodo, este notificaría el valor 5 para la métrica, incluso aunque el cliente solo realizara una solicitud. Estadísticas de nodo pertinentes: promedio Estadísticas de clúster pertinentes: suma, máximo, promedio
`WarmConcurrentSearchLatency`	La diferencia en el tiempo total, en milisegundos, calculada por todas las búsquedas que usan la búsqueda de segmentos simultánea en un nodo de UltraWarm entre el minuto N y el minuto (N-1). Estadísticas de nodo pertinentes: promedio Estadísticas de clúster pertinentes: máximo, promedio

Métrica	Descripción
`WarmThreadpoolIndexSearcherQueue`	El número de tareas en cola en el grupo de subprocesos del buscador de índices de UltraWarm. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: suma, máximo, promedio
`WarmThreadpoolIndexSearcherRejected`	El número de tareas rechazadas en el grupo de subprocesos del buscador de índices de UltraWarm. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: suma
`WarmThreadpoolIndexSearcherThreads`	El tamaño del grupo de subprocesos del buscador de índices de UltraWarm. Estadísticas de nodo pertinentes: máximo Estadísticas de clúster pertinentes: suma, promedio

Métricas de almacenamiento en frío

Amazon OpenSearch Service ofrece las siguientes métricas para el almacenamiento frío.

Métrica	Descripción
`ColdStorageSpaceUtilization`	La cantidad total de espacio de almacenamiento en frío, en MiB, que el clúster utiliza. Estadísticas pertinentes: Max
`ColdToWarmMigrationFailureCount`	El número total de errores de migración del almacenamiento en frío al almacenamiento templado. Estadísticas pertinentes: suma

Métrica	Descripción
`ColdToWarmMigrationLatency`	Cantidad de tiempo para que se completen de forma correcta las migraciones del almacenamiento en frío al almacenamiento templado. Estadísticas pertinentes: promedio
`ColdToWarmMigrationQueueSize`	Número de índices a la espera actualmente para migrar del almacenamiento frío al almacenamiento templado. Estadísticas pertinentes: máximo
`ColdToWarmMigrationSuccessCount`	El número total de migraciones exitosas del almacenamiento en frío al almacenamiento templado. Estadísticas pertinentes: suma
`WarmToColdMigrationFailureCount`	El número total de errores de migración del almacenamiento templado al almacenamiento en frío. Estadísticas pertinentes: suma
`WarmToColdMigrationLatency`	Cantidad de tiempo para que se completen de forma correcta las migraciones del almacenamiento templado al almacenamiento en frío. Estadísticas pertinentes: promedio
`WarmToColdMigrationQueueSize`	Número de índices a la espera actualmente para migrar del almacenamiento templado al almacenamiento frío. Estadísticas pertinentes: máximo
`WarmToColdMigrationSuccessCount`	El número total de migraciones exitosas del almacenamiento templado al almacenamiento en frío. Estadísticas pertinentes: suma

Métricas de instancias optimizadas de OpenSearch (OR1)

Amazon OpenSearch Service ofrece las siguientes métricas para las instancias OR1.

Métrica	Descripción
`RemoteStorageUsedSpace`	Cantidad total del espacio de Amazon S3, en MiB, que utiliza el clúster. Estadísticas pertinentes: suma
`RemoteStorageWriteRejected`	El número total de solicitudes rechazadas en las particiones principales debido a la presión de replicación y almacenamiento remoto. Se calcula a partir del último inicio del proceso de OpenSearch Service. Estadísticas pertinentes: suma
`ReplicationLagMaxTime`	El tiempo, en milisegundos, que las particiones de la réplica permanecen detrás de las particiones primarias. Estadísticas pertinentes: máximo

Métricas de alertas

Amazon OpenSearch Service ofrece las siguientes métricas para alertas.

Métrica	Descripción
`AlertingDegraded`	Un valor de 1 significa que el índice de alerta es rojo o que uno o más nodos no van según lo programado. Un valor de 0 indica un comportamiento normal. Estadísticas pertinentes: máximo
`AlertingIndexExists`	Un valor de 1 significa que el índice `.opensearch-alerting-config` existe. Un valor de 0 significa que no existe. Hasta que utilice la característica de alerta por primera vez, este valor sigue siendo 0.

Métrica	Descripción
	Estadísticas pertinentes: máximo
`AlertingIndexStatus.green`	El estado de salud del índice. Un valor de 1 significa verde. Un valor de 0 significa que el índice no existe o no es verde. Estadísticas pertinentes: máximo
`AlertingIndexStatus.red`	El estado de salud del índice. Un valor de 1 significa rojo. Un valor de 0 significa que el índice no existe o no es rojo. Estadísticas pertinentes: máximo
`AlertingIndexStatus.yellow`	El estado de salud del índice. Un valor de 1 significa amarillo. Un valor de 0 significa que el índice no existe o no es amarillo. Estadísticas pertinentes: máximo
`AlertingNodesNotOnSchedule`	Un valor de 1 significa que algunos trabajos no se están ejecutando según lo programado. Un valor de 0 significa que todos los trabajos de alerta se están ejecutando según lo programado (o que no hay trabajos de alerta). Verifique la consola de OpenSearch Service o realice una solicitud `_nodes/stats` para saber si algún nodo muestra un alto consumo de recursos. Estadísticas pertinentes: máximo
`AlertingNodesOnSchedule`	Un valor de 1 significa que todos los trabajos de alerta se están ejecutando según lo programado (o que no hay trabajos de alerta). Un valor de 0 significa que algunos trabajos no se están ejecutando según lo programado. Estadísticas pertinentes: máximo

Métrica	Descripción
AlertingScheduledJobEnabled	Un valor de 1 significa que la configuración del clúster opensearch.scheduled_jobs.enabled es verdadera. Un valor de 0 significa que es falsa y que los trabajos programados están deshabilitados. Estadísticas pertinentes: máximo

Métricas de detección de anomalías

Amazon OpenSearch Service ofrece las siguientes métricas para la detección de anomalías.

Métrica	Descripción
ADPluginUnhealthy	El valor 1 significa que el complemento de detección de anomalías no funciona correctamente, bien debido a un gran número de errores o bien porque uno de los índices que utiliza es rojo. Un valor de 0 indica que el complemento funciona como se esperaba. Estadísticas pertinentes: máximo
ADExecuteRequestCount	El número de solicitudes para detectar anomalías. Estadísticas pertinentes: suma
ADExecuteFailureCount	El número de solicitudes erróneas para detectar anomalías. Estadísticas pertinentes: suma
ADHCExecuteFailureCount	El número de errores de solicitud al detectar anomalías para los detectores de alta cardinalidad. Estadísticas pertinentes: suma
ADHCExecuteRequestCount	El número de solicitudes al detectar anomalías para detectores de alta cardinalidad. Estadísticas pertinentes: suma

Métrica	Descripción
`ADAnomalyResultsIndexStatusIndexExists`	Un valor de 1 significa que existe el índice al que apunta el alias `.opensearch-anomaly-results`. Hasta que no se utilice la detección de anomalías por primera vez, este valor permanece en 0. Estadísticas pertinentes: máximo
`ADAnomalyResultsIndexStatus.red`	Un valor de 1 significa que el índice al que apunta el alias `.opensearch-anomaly-results` es rojo. Un valor de 0 significa que no lo es. Hasta que no se utilice la detección de anomalías por primera vez, este valor permanece en 0. Estadísticas pertinentes: máximo
`ADAnomalyDetectorsIndexStatusIndexExists`	Un valor de 1 significa que el índice `.opensearch-anomaly-detectors` existe. Un valor de 0 significa que no existe. Hasta que no se utilice la detección de anomalías por primera vez, este valor permanece en 0. Estadísticas pertinentes: máximo
`ADAnomalyDetectorsIndexStatus.red`	Un valor de 1 significa que el índice `.opensearch-anomaly-detectors` es rojo. Un valor de 0 significa que no lo es. Hasta que no se utilice la detección de anomalías por primera vez, este valor permanece en 0. Estadísticas pertinentes: máximo
`ADModelsCheckpointIndexStatusIndexExists`	Un valor de 1 significa que el índice `.opensearch-anomaly-checkpoints` existe. Un valor de 0 significa que no existe. Hasta que no se utilice la detección de anomalías por primera vez, este valor permanece en 0. Estadísticas pertinentes: máximo

Métrica	Descripción
`ADModelsCheckpointIndexStatus.red`	Un valor de 1 significa que el índice `.opensearch-anomaly-checkpoints` es rojo. Un valor de 0 significa que no lo es. Hasta que no se utilice la detección de anomalías por primera vez, este valor permanece en 0. Estadísticas pertinentes: máximo

Métricas de búsqueda asíncrona

Amazon OpenSearch Service ofrece las siguientes métricas para la búsqueda asíncrona.

Estadísticas del nodo coordinador de búsqueda asíncrona (por nodo coordinador)

Métrica	Descripción
`AsynchronousSearchSubmissionRate`	El número de búsquedas asíncronas enviadas en el último minuto.
`AsynchronousSearchInitializedRate`	El número de búsquedas asíncronas inicializadas en el último minuto.
`AsynchronousSearchRunningCurrent`	El número de búsquedas asíncronas que se ejecutan actualmente.
`AsynchronousSearchCompletionRate`	El número de búsquedas asíncronas realizadas correctamente en el último minuto.
`AsynchronousSearchFailureRate`	El número de búsquedas asíncronas que se completaron y fallaron en el último minuto.

Métrica	Descripción
AsynchronousSearchPersistRate	El número de búsquedas asíncronas que persistieron en el último minuto.
AsynchronousSearchPersistFailedRate	El número de búsquedas asíncronas que no pudieron persistir en el último minuto.
AsynchronousSearchRejected	El número total de búsquedas asíncronas rechazadas desde el tiempo de actividad del nodo.
AsynchronousSearchCancelled	El número total de búsquedas asíncronas canceladas desde el tiempo de actividad del nodo.
AsynchronousSearchMaxRunningTime	Duración de la búsqueda asíncrona de ejecución más larga en un nodo en el último minuto.

Estadísticas del clúster de búsqueda asíncrona

Métrica	Descripción
AsynchronousSearchStoreHealth	El estado de la tienda en el índice persistente (ROJO/no ROJO) en el último minuto.
AsynchronousSearchStoreSize	El tamaño del índice del sistema en todas las particiones en el último minuto.
AsynchronousSearch	El número de respuestas almacenadas en el índice del sistema en el último minuto.

Métrica	Descripción
StoredResponseCount	

Métricas de ajuste automático

Amazon OpenSearch Service ofrece las siguientes métricas de ajuste automático.

Métrica	Descripción
AutoTuneChangesHistoryHeapSize	El historial de cambios en MiB para los valores de ajuste del tamaño del montón.
AutoTuneChangesHistoryJVMYoungGenArgs	El historial de cambios de los argumentos de JVM YongGen.
AutoTuneFailed	Un valor booleano que indica si el cambio de ajuste automático produjo un error.
AutoTuneSucceeded	Un valor booleano que indica si el cambio de ajuste automático se realizó correctamente.
AutoTuneValue	El historial de cambios de cola (recuento) y el historial de cambios de ajustes de caché (en MiB) para los cambios no disruptivos.

Métricas de Multi-AZ con modo de espera

Amazon OpenSearch Service ofrece las siguientes métricas para Multi-AZ con modo de espera.

Métricas a nivel de nodo para los nodos de datos en zonas de disponibilidad activas

Métrica	Descripción
`CPUUtilization`	El porcentaje de uso de CPU para los nodos de datos del clúster. Máximo muestra el nodo con el mayor uso de CPU. Promedio represent a todos los nodos del clúster. Esta métrica también está disponible para los nodos individuales.
`FreeStorageSpace`	El espacio libre para los nodos de datos del clúster. `Sum` muestra el espacio libre total del clúster, pero debe dejar el periodo en un minuto para recibir un valor preciso. `Minimum` y `Maximum` muestran los nodos con el mínimo y máximo espacio libre, respectivamente. Esta métrica también está disponible para los nodos individuales. OpenSearch Service muestra una `ClusterBlockException` cuando esta métrica alcanza `0`. Para efectuar la recuperación, debe eliminar índices, agregar instancias de mayor tamaño o bien agregar almacenamiento basado en EBS a las instancias existentes. Para más información, consulte the section called “Falta de espacio de almacenamiento disponible”. La consola de OpenSearch Service muestra este valor en GiB. La consola de Amazon CloudWatch lo muestra en MiB.
`JVMMemoryPressure`	El porcentaje máximo del montón de Java que se emplea para todos los nodos de datos del clúster. OpenSearch Service utiliza la mitad de la RAM de una instancia para la pila de Java, hasta un tamaño de 32 GiB. Puede escalar las instancias verticalmente hasta 64 GiB de RAM y después escalarlas horizontalmente mediante el agregado de instancias. Consulte the section called “Alarmas de CloudWatch recomendadas”.
`SysMemoryUtilization`	El porcentaje de memoria de la instancia que está en uso. Es normal obtener valores altos para esta métrica y, por lo general, no representan un problema con el clúster. Para obtener un mejor indicador de posibles problemas de rendimiento y estabilidad, consulte la métrica `JVMMemoryPressure` .

Métrica	Descripción
IndexingLatency	La diferencia en el tiempo total, en milisegundos, calculada por todas las operaciones de indexación en un nodo entre el minuto N y el minuto (N-1).
IndexingRate	El número de operaciones de indexación por minuto.
SearchLatency	La diferencia en el tiempo total, en milisegundos, calculada por todas las búsquedas en un nodo entre el minuto N y el minuto (N-1).
SearchRate	El número total de peticiones de búsqueda por minuto de todas las particiones de un nodo de datos.
ThreadpoolSearchQueue	El número de tareas en cola en el grupo de subprocesos de búsqueda. Si el tamaño de la cola es sistemáticamente elevado, considere la posibilidad de escalar el clúster. El tamaño máximo de la cola de búsqueda es 1000.
ThreadpoolWriteQueue	El número de tareas en cola en el grupo de subprocesos de escritura.
ThreadpoolSearchRejected	El número de tareas rechazadas en el grupo de subprocesos de búsqueda. Si este número crece continuamente, considere la posibilidad de escalar el clúster.
ThreadpoolWriteRejected	El número de tareas rechazadas en el grupo de subprocesos de escritura.

Métricas de nivel de clúster para clústeres en zonas de disponibilidad activas

Métrica	Descripción
DataNodes	El número total de particiones activas y en espera.
DataNodesShards.active	El número total de particiones primarias y de réplicas activas.

Métrica	Descripción
`DataNodes Shards.un assigned`	El número de particiones que no se asignaron a los nodos del clúster.
`DataNodes Shards.in itializing`	El número de particiones que se encuentran en inicialización.
`DataNodes Shards.re locating`	El número de particiones que se encuentran en reubicación.

Métricas de rotación de zonas de disponibilidad

Si `ActiveReads.`*`Availability-Zone`* `= 1`, entonces la zona está activa. Si `ActiveReads.`*`Availability-Zone`* `= 0`, entonces la zona está en espera.

Métricas de un momento dado

Amazon OpenSearch Service ofrece las siguientes métricas para la búsqueda en un momento dado (PIT).

Estadísticas del nodo coordinador de PIT (por nodo coordinador)

Métrica	Descripción
`CurrentPo intInTime`	El número de contextos de búsqueda PIT activos en el nodo.
`TotalPoin tInTime`	El número de contextos de búsqueda PIT caducados desde el tiempo de actividad del nodo.
`AvgPointI nTimeAliveTime`	El keep-alive promedio de los contextos de búsqueda PIT desde el tiempo de actividad del nodo.

Métrica	Descripción
`HasActive PointInTime`	Un valor de 1 indica que hay contextos PIT activos en los nodos desde el tiempo de actividad del nodo. Un valor de 0 indica que no hay.
`HasUsedPo intInTime`	Un valor de 1 indica que hay contextos PIT vencidos en los nodos desde el tiempo de actividad del nodo. Un valor de 0 indica que no hay.

Métricas de SQL

Amazon OpenSearch Service ofrece las siguientes métricas para la compatibilidad con SQL.

Métrica	Descripción
`SQLFailed RequestCo untByCusErr`	El número de solicitudes a la API `_sql` que no se han realizado correctamente debido a un problema del cliente. Por ejemplo, una solicitud podría devolver el código de estado HTTP 400 debido a una excepción `IndexNotFoundException` . Estadísticas pertinentes: suma
`SQLFailed RequestCo untBySysErr`	El número de solicitudes a la API `_sql` que no se han realizado correctamente debido a un problema del servidor o a una limitación de características. Por ejemplo, una solicitud podría devolver el código de estado HTTP 503 debido a una excepción `VerificationExcept ion` . Estadísticas pertinentes: suma
`SQLRequestCount`	El número de solicitudes a la API `_sql`. Estadísticas pertinentes: suma
`SQLDefaul tCursorRe questCount`	Similar a `SQLRequestCount` , pero solo cuenta las solicitudes de paginación. Estadísticas pertinentes: suma

Métrica	Descripción
SQLUnhealthy	Un valor de 1 indica que, en respuesta a ciertas solicitudes, el complemento SQL devuelve códigos de respuesta 5xx o pasa DSL de consulta no válida a OpenSearch. Las demás solicitudes deberían realizarse correctamente. Un valor de 0 indica que no hay errores recientes. Si aparece sistemáticamente un valor de 1, solucione los problemas en las solicitudes que sus clientes están realizando al complemento. Estadísticas pertinentes: máximo

Métricas k-NN

Amazon OpenSearch Service incluye las siguientes métricas para el complemento k-nearest neighbor (k-NN).

Métrica	Descripción
KNNCacheCapacityReached	Métrica por nodo para saber si se alcanzó la capacidad de caché. Esta métrica solo es pertinente para la búsqueda aproximada de k-NN. Estadísticas pertinentes: máximo
KNNCircuitBreakerTriggered	Métrica por clúster para saber si se activa el disyuntor. Si algún nodo devuelve un valor de 1 para KNNCacheCapacityReached , este valor también devolverá 1. Esta métrica solo es pertinente para la búsqueda aproximada de k-NN. Estadísticas pertinentes: máximo
KNNEvictionCount	Métrica por nodo para el número de gráficos que se expulsaron de la caché debido a limitaciones de memoria o tiempo de inactividad. Las expulsiones explícitas que se producen debido a la eliminación de indexación no se cuentan. Esta métrica solo es pertinente para la búsqueda aproximada de k-NN.

Métrica	Descripción
	Estadísticas pertinentes: suma
`KNNGraphIndexErrors`	Métrica por nodo para el número de solicitudes para agregar el campo `knn_vector` de un documento a un gráfico que produjo un error. Estadísticas pertinentes: suma
`KNNGraphIndexRequests`	Métrica por nodo para el número de solicitudes para agregar el campo `knn_vector` de un documento a un gráfico. Estadísticas pertinentes: suma
`KNNGraphMemoryUsage`	Métrica por nodo para el tamaño actual de caché (tamaño total de todos los gráficos en memoria) en kilobytes. Esta métrica solo es pertinente para la búsqueda aproximada de k-NN. Estadísticas pertinentes: promedio
`KNNGraphQueryErrors`	Métrica por nodo para el número de consultas de gráficos que produjeron un error. Estadísticas pertinentes: suma
`KNNGraphQueryRequests`	Métrica por nodo para el número de consultas de gráficos. Estadísticas pertinentes: suma
`KNNHitCount`	Métrica por nodo para el número de aciertos de caché. Un acierto de caché se produce cuando un usuario consulta un gráfico que ya está cargado en la memoria. Esta métrica solo es pertinente para la búsqueda aproximada de k-NN. Estadísticas pertinentes: suma

Métrica	Descripción
KNNLoadExceptionCount	Métrica por nodo para el número de veces que se produjo una excepción al intentar cargar un gráfico en la caché. Esta métrica solo es pertinente para la búsqueda aproximada de k-NN. Estadísticas pertinentes: suma
KNNLoadSuccessCount	Métrica por nodo para el número de veces que el complemento cargó correctamente un gráfico en la caché. Esta métrica solo es pertinente para la búsqueda aproximada de k-NN. Estadísticas pertinentes: suma
KNNMissCount	Métrica por nodo para el número de errores de caché. Un error de caché se produce cuando un usuario consulta un gráfico que aún no está cargado en la memoria. Esta métrica solo es pertinente para la búsqueda aproximada de k-NN. Estadísticas pertinentes: suma
KNNQueryRequests	Métrica por nodo para el número de solicitudes de consulta que recibió el complemento k-NN. Estadísticas pertinentes: suma
KNNScriptCompilationErrors	Métrica por nodo para el número de errores durante la compilación del script. Esta estadística solo es relevante para la búsqueda de scripts de puntuación k-NN. Estadísticas pertinentes: suma
KNNScriptCompilations	Métrica por nodo para el número de veces que se ha compilado el script k-NN. Por lo general, este valor debe ser 1 o 0, pero si se llena la caché que contiene los scripts compilados, el script k-NN podría ser recompilado. Esta estadística solo es relevante para la búsqueda de scripts de puntuación k-NN. Estadísticas pertinentes: suma

Métrica	Descripción
KNNScriptQueryErrors	Métrica por nodo para el número de errores durante las consultas de scripts. Esta estadística solo es relevante para la búsqueda de scripts de puntuación k-NN. Estadísticas pertinentes: suma
KNNScriptQueryRequests	Métrica por nodo para el número total de consultas de script. Esta estadística solo es relevante para la búsqueda de scripts de puntuación k-NN. Estadísticas pertinentes: suma
KNNTotalLoadTime	El tiempo en nanosegundos que k-NN tardó en cargar gráficos en la caché. Esta métrica solo es pertinente para la búsqueda aproximada de k-NN. Estadísticas pertinentes: suma

Métricas de búsqueda entre clústeres

Amazon OpenSearch Service ofrece las siguientes métricas para la búsqueda entre clústeres.

Métricas de dominio de origen

Métrica	Dimensión	Descripción
CrossClusterOutboundConnections	ConnectionId	Número de nodos conectados. Si la respuesta incluye uno o más dominios omitidos, utilice esta métrica para rastrear las conexiones que no estén en buen estado. Si este número cae a 0, entonces la conexión no está en buen estado.
CrossClusterOutboundRequests	ConnectionId	El número de peticiones de búsqueda enviadas al dominio de destino. Utilícelo para comprobar si la carga de peticiones de búsqueda en clústeres está sobrecargando su dominio, correlacione cualquier pico en esta métrica con cualquier pico JVM/CPU.

Métrica de dominio de destino

Métrica	Dimensión	Descripción
`CrossClusterInboundRequests`	`ConnectionId`	El número de solicitudes de conexión entrantes recibidas desde el dominio de origen.

Agregue una alarma de CloudWatch en caso de que pierda una conexión de forma inesperada. Para ver los pasos para crear una alarma, consulte Crear una alarma de CloudWatch basada en un umbral estático.

Métricas de replicación entre clústeres

Amazon OpenSearch Service ofrece las siguientes métricas para la replicación entre clústeres.

Métrica	Descripción
`ReplicationRate`	La tasa media de operaciones de replicación por segundo. Esta métrica es similar a `IndexingRate` .
`LeaderCheckPoint`	Para lograr una conexión específica, la suma de los valores de punto de control líder entre todos los índices de replicación. Puede utilizar esta métrica para medir la latencia de replicación.
`FollowerCheckPoint`	Para lograr una conexión específica, la suma de los valores de punto de control seguidor entre todos los índices de replicación. Puede utilizar esta métrica para medir la latencia de replicación.
`ReplicationNumSyncingIndices`	El número de índices que tienen un estado de replicación de `SYNCING`.
`ReplicationNumBootstrappingIndices`	El número de índices que tienen un estado de replicación de `BOOTSTRAPPING` .

Métrica	Descripción
`ReplicationNumPausedIndices`	El número de índices que tienen un estado de replicación de `PAUSED`.
`ReplicationNumFailedIndices`	El número de índices que tienen un estado de replicación de `FAILED`.
`CrossClusterOutboundReplicationRequests`	El número de solicitudes de transporte de replicación en el dominio seguidor. Las solicitudes de transporte son internas y se producen cada vez que se llama una operación de la API de replicación. También se producen cuando el dominio seguidor sondea los cambios con respecto al dominio líder.
`CrossClusterInboundReplicationRequests`	El número de solicitudes de transporte de replicación en el dominio principal. Las solicitudes de transporte son internas y se producen cada vez que se llama una operación de API de replicación.
`AutoFollowNumSuccessStartReplication`	El número de índices de seguidores que ha creado correctamente una regla de replicación para una conexión específica.
`AutoFollowNumFailedStartReplication`	El número de índices de seguidores que no pudo crear una regla de replicación cuando había un patrón coincidente. Este problema puede deberse a un problema de red en el clúster remoto o a un problema de seguridad (es decir, el rol asociado no tiene permiso para iniciar la replicación).
`AutoFollowLeaderCallFailure`	Si ha habido alguna consulta fallida del índice de seguidores al índice de líderes para obtener nuevos datos. Un valor de 1 significa que ha habido 1 o más llamadas fallidas en el último minuto.

Aprender a clasificar métricas

Amazon OpenSearch Service ofrece las siguientes métricas para aprender a clasificar.

Métrica	Descripción
`LTRRequestTotalCount`	Recuento total de solicitudes de clasificación.
`LTRRequestErrorCount`	Recuento total de solicitudes fallidas.
`LTRStatus.red`	Comprueba si uno de los índices necesarios para ejecutar el complemento es rojo.
`LTRMemoryUsage`	Memoria total utilizada por el complemento.
`LTRFeatureMemoryUsageInBytes`	La cantidad de memoria, en bytes, utilizada por los campos de características de Learning to Rank (Aprender a clasificar).
`LTRFeaturesetMemoryUsageInBytes`	La cantidad de memoria, en bytes, que utiliza todos los conjuntos de características de Learning to Rank (Aprender a clasificar).
`LTRModelMemoryUsageInBytes`	La cantidad de memoria, en bytes, utilizada por todos los modelos Learning to Rank (Aprender a clasificar).

Métricas del lenguaje de procesamiento de canalizaciones

Amazon OpenSearch Service ofrece las siguientes métricas para el lenguaje de procesamiento de canalizaciones.

Métrica	Descripción
`PPLFailedRequestCountByCusErr`	El número de solicitudes a la API `_ppl` que no se han realizado correctamente debido a un problema del cliente. Por ejemplo, una solicitud podría devolver el código de estado HTTP 400 debido a una excepción `IndexNotFoundException` .
`PPLFailedRequestCountBySysErr`	El número de solicitudes a la API `_ppl` que no se han realizado correctamente debido a un problema del servidor o a una limitación de características. Por ejemplo, una solicitud podría devolver el código de estado HTTP 503 debido a una excepción `VerificationException` .
`PPLRequestCount`	El número de solicitudes a la API `_ppl`.

Monitoreo de registros de OpenSearch con Registros de Amazon CloudWatch

Amazon OpenSearch Service expone los siguientes registros de OpenSearch a través de Registros de Amazon CloudWatch:

- Registros de errores
- Registros lentos de solicitudes de búsqueda
- Registros lentos de particiones
- Registros de auditoría

Los registros lentos de particiones de búsqueda, los registros lentos de particiones de índice y los registros de errores son útiles para la resolución de problemas de rendimiento y estabilidad. Los registros de auditoría realizan un seguimiento de la actividad de los usuarios con fines de conformidad. Todos los registros están deshabilitados de forma predeterminada. Si están habilitados, se aplica el precio estándar de CloudWatch.

> **Note**
>
> Los registros de errores solo están disponibles para la versión 5.1 de OpenSearch y Elasticsearch y posteriores. Los registros lentos están disponibles en todas las versiones de OpenSearch y Elasticsearch.

Para los registros, OpenSearch utiliza Apache Log4j 2 y sus niveles de registro integrados (de menor a mayor gravedad) `TRACE`, `DEBUG`, `INFO`, `WARN`, `ERROR` y `FATAL`.

Si habilita registros de errores, OpenSearch Service publica líneas de registro de `WARN`, `ERROR` y `FATAL` en CloudWatch. OpenSearch Service también publica varias excepciones del nivel `DEBUG`, incluido lo siguiente:

- `org.opensearch.index.mapper.MapperParsingException`
- `org.opensearch.index.query.QueryShardException`
- `org.opensearch.action.search.SearchPhaseExecutionException`
- `org.opensearch.common.util.concurrent.OpenSearchRejectedExecutionException`
- `java.lang.IllegalArgumentException`

Los registros de errores ayudan a solucionar problemas en muchas situaciones, entre las que se incluyen las siguientes:

- Problemas de compilación de script Painless
- Consultas no válidas
- Problemas de indexación
- Errores de instantáneas
- Fallas de migración de la administración de estados de índice

> **Note**
>
> No todos los errores se registran en los registros de errores.

> **Note**
>
> OpenSearch Service no registra todos los errores que se producen.

Temas

- Habilitación de la publicación de registros (consola)
- Habilitación de la publicación de registros (AWS CLI)
- Habilitación de la publicación de registros (SDK de AWS)
- Habilitar publicación de registros (CloudFormation)
- Establecimiento de umbrales de registros lentos de solicitudes de búsqueda
- Establecimiento de umbrales de registros lentos de particiones
- Prueba de registros lentos
- Visualización de registros

Habilitación de la publicación de registros (consola)

La consola de OpenSearch Service es la manera más sencilla de habilitar la publicación de registros en CloudWatch.

Para habilitar la publicación de registros en CloudWatch (consola)

1. Vaya a aws.amazon.com, luego elija Iniciar sesión y proporcione sus credenciales.
2. En Análisis, seleccione Amazon OpenSearch Service.
3. Seleccione el dominio que desea actualizar.
4. En la pestaña Registros, seleccione un tipo de registro y luego elija Habilitar.
5. Cree un grupo de registros de CloudWatch o elija uno existente.

   > **Note**
   >
   > Si prevé habilitar varios registros, recomendamos publicar cada uno en su propio grupo de registros. Esta separación hace que los registros sean más fáciles de analizar.

6. Seleccione una política de acceso que contenga los permisos adecuados o cree una política mediante el JSON que proporciona la consola:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "es.amazonaws.com"
      },
      "Action": [
        "logs:PutLogEvents",
        "logs:CreateLogStream"
      ],
      "Resource": "arn:aws:logs:us-east-1:111122223333:log-
group:cw_log_group_name:*"
    }
  ]
}
```

Le recomendamos que agregue las claves de condición `aws:SourceAccount` y `aws:SourceArn` para protegerse contra la política problema del suplente confuso. La cuenta de origen es la propietaria del dominio y el ARN de origen es el ARN del dominio. Su dominio debe estar en el software de servicio R20211203 o posterior para agregar estas claves de condición.

Por ejemplo, podría agregar el siguiente bloque de condición a la política:

```
"Condition": {
    "StringEquals": {
        "aws:SourceAccount": "account-id"
    },
    "ArnLike": {
        "aws:SourceArn": "arn:aws:es:region:account-id:domain/domain-name"
    }
}
```

> **Important**
>
> Registros de CloudWatch admiten 10 políticas de recursos por región. Si planea habilitar los registros para varios dominios de OpenSearch Service, debe crear y reutilizar una política más amplia que incluya varios grupos de registros para evitar alcanzar este límite. Para obtener información sobre los pasos para actualizar la política, consulte the section called "Habilitación de la publicación de registros (AWS CLI)".

7. Seleccione Habilitar.

   El estado del dominio cambia de Activo a En proceso. El estado debe volver a Activo antes de que la publicación de registro esté habilitada. Este cambio suele tardar 30 minutos, pero puede tardar más en función de la configuración del dominio.

Si ha habilitado uno de los registros lentos de particiones, consulte the section called "Establecimiento de umbrales de registros lentos de particiones". Si habilitó registros de auditorías, consulte the section called "Paso 2: Activar los registros de auditoría en OpenSearch Dashboards". Si habilitó únicamente los registros de errores, no tiene que realizar ningún paso de configuración adicional.

Habilitación de la publicación de registros (AWS CLI)

Para poder habilitar la publicación de registros, necesita un grupo de registros de CloudWatch Logs. Si aún no lo tiene, puede crear uno mediante el siguiente comando:

```
aws logs create-log-group --log-group-name my-log-group
```

Introduzca el siguiente comando para encontrar el ARN del grupo de registros y, a continuación, anótelo:

```
aws logs describe-log-groups --log-group-name my-log-group
```

Ahora puede dar permisos a OpenSearch Service para escribir en el grupo de registros. Debe proporcionar el ARN del grupo de registros cerca del final del comando:

```
aws logs put-resource-policy \
  --policy-name my-policy \
```

```
  --policy-document '{ "Version": "2012-10-17",        "Statement": [{ "Sid":
 "", "Effect": "Allow", "Principal": { "Service": "es.amazonaws.com"}, "Action":
[ "logs:PutLogEvents","logs:CreateLogStream"],"Resource": "cw_log_group_arn:*"}]}'
```

> **Important**
>
> Registros de CloudWatch admiten 10 políticas de recursos por región. Si planea habilitar registros lentos de partición para varios dominios de OpenSearch Service, debe crear y reutilizar una política más amplia que incluya varios grupos de registros para evitar alcanzar este límite.

Si tiene que revisar esta política más adelante, utilice el comando `aws logs describe-resource-policies`. Para actualizar la política, ejecute el mismo comando `aws logs put-resource-policy` con un nuevo documento de política.

Finalmente, puede utilizar la opción `--log-publishing-options` para habilitar la publicación. La sintaxis de la opción es la misma para los comandos `create-domain` y `update-domain-config`.

Parámetro	Valores válidos
`--log-publishing-options`	`SEARCH_SLOW_LOGS={CloudWatchLogsLogGroupArn= cw_log_group_arn ,Enabled=true|false}`
	`INDEX_SLOW_LOGS={CloudWatchLogsLogGroupArn= cw_log_group_arn ,Enabled=true|false}`
	`ES_APPLICATION_LOGS={CloudWatchLogsLogGroupArn= cw_log_group_arn ,Enabled=true|false}`
	`AUDIT_LOGS={CloudWatchLogsLogGroupArn= cw_log_group_arn ,Enabled=true|false}`

> **Note**
>
> Si prevé habilitar varios registros, recomendamos publicar cada uno en su propio grupo de registros. Esta separación hace que los registros sean más fáciles de analizar.

Ejemplo

En el siguiente ejemplo se habilita la publicación de los registros lentos de particiones de búsqueda e índice para el dominio especificado:

```
aws opensearch update-domain-config \
  --domain-name my-domain \
  --log-publishing-options
 "SEARCH_SLOW_LOGS={CloudWatchLogsLogGroupArn=arn:aws:logs:us-east-1:123456789012:log-group:my-log-
group,Enabled=true},INDEX_SLOW_LOGS={CloudWatchLogsLogGroupArn=arn:aws:logs:us-east-1:123456789012:log-group:my-other-log-group,Enabled=true}"
```

Para deshabilitar la publicación en CloudWatch, ejecute el mismo comando con `Enabled=false`.

Si ha habilitado uno de los registros lentos de particiones, consulte the section called "Establecimiento de umbrales de registros lentos de particiones". Si habilitó registros de auditorías, consulte the section called "Paso 2: Activar los registros de auditoría en OpenSearch Dashboards". Si habilitó únicamente los registros de errores, no tiene que realizar ningún paso de configuración adicional.

Habilitación de la publicación de registros (SDK de AWS)

Para habilitar la publicación de registros, primero debe crear un grupo de registros de CloudWatch, obtener su ARN y otorgar permisos a OpenSearch Service para escribir en ella. Las operaciones pertinentes se documentan en la Referencia de la API de Amazon CloudWatch Logs:

- `CreateLogGroup`
- `DescribeLogGroup`
- `PutResourcePolicy`

Puede obtener acceso a estas operaciones mediante los SDK de AWS.

Los AWS SDK (excepto los de Android e iOS SDK) admiten todas las operaciones definidas en la Referencia de la API de Amazon OpenSearch Service, incluida la opción `--log-publishing-options` para `CreateDomain` y `UpdateDomainConfig`.

Si ha habilitado uno de los registros lentos de particiones, consulte the section called "Establecimiento de umbrales de registros lentos de particiones". Si habilitó únicamente los registros de errores, no tiene que realizar ningún paso de configuración adicional.

Habilitar publicación de registros (CloudFormation)

En este ejemplo, se utiliza CloudFormation para crear un grupo de registros llamado `opensearch-logs`, se asignan los permisos adecuados y, a continuación, se crea un dominio con la publicación de registros habilitada para registros de aplicaciones, registros lentos de particiones de búsqueda y registros lentos de índice.

Para poder habilitar la publicación de registros, hay que crear un grupo de registros de CloudWatch:

```
Resources:
  OpenSearchLogGroup:
    Type: AWS::Logs::LogGroup
    Properties:
      LogGroupName: opensearch-logs
Outputs:
  Arn:
    Value:
      'Fn::GetAtt':
        - OpenSearchLogGroup
        - Arn
```

La plantilla genera el ARN del grupo de registros. En este caso, el ARN es `arn:aws:logs:us-east-1:123456789012:log-group:opensearch-logs`.

Mediante el ARN, cree una política de recursos que dé permisos a OpenSearch Service para escribir en el grupo de registros:

```
Resources:
 OpenSearchLogPolicy:
   Type: AWS::Logs::ResourcePolicy
   Properties:
     PolicyName: my-policy
     PolicyDocument: "{ \"Version\": \"2012-10-17\", \"Statement\": [{ \"Sid\": \"\",
 \"Effect\": \"Allow\", \"Principal\": { \"Service\": \"es.amazonaws.com\"}, \"Action
\":[ \"logs:PutLogEvents\",\"logs:CreateLogStream\"],\"Resource\": \"arn:aws:logs:us-
east-1:123456789012:log-group:opensearch-logs:*\"}]}"
```

Por último, cree la siguiente pila de CloudFormation, que genera un dominio de OpenSearch Service con publicación de registros. La política de acceso permite al usuario de la Cuenta de AWS realizar todas las solicitudes HTTP al dominio.

```
Resources:
  OpenSearchServiceDomain:
    Type: "AWS::OpenSearchService::Domain"
    Properties:
      DomainName: my-domain
      EngineVersion: "OpenSearch_1.0"
      ClusterConfig:
        InstanceCount: 2
        InstanceType: "r6g.xlarge.search"
        DedicatedMasterEnabled: true
        DedicatedMasterCount: 3
        DedicatedMasterType: "r6g.xlarge.search"
      EBSOptions:
        EBSEnabled: true
        VolumeSize: 10
        VolumeType: "gp2"
      AccessPolicies:
        Version: "2012-10-17"
        Statement:
            Effect: "Allow"
            Principal:
                AWS: "arn:aws:iam::123456789012:user/es-user"
            Action: "es:*"
            Resource: "arn:aws:es:us-east-1:123456789012:domain/my-domain/*"
      LogPublishingOptions:
        ES_APPLICATION_LOGS:
          CloudWatchLogsLogGroupArn: "arn:aws:logs:us-east-1:123456789012:log-
group:opensearch-logs"
          Enabled: true
        SEARCH_SLOW_LOGS:
          CloudWatchLogsLogGroupArn: "arn:aws:logs:us-east-1:123456789012:log-
group:opensearch-logs"
          Enabled: true
        INDEX_SLOW_LOGS:
          CloudWatchLogsLogGroupArn: "arn:aws:logs:us-east-1:123456789012:log-
group:opensearch-logs"
          Enabled: true
```

Para obtener información detallada sobre la sintaxis, consulte las opciones de publicación de registros en la Guía del usuario de CloudFormation.

Establecimiento de umbrales de registros lentos de solicitudes de búsqueda

Los registros lentos de solicitudes de búsqueda están disponibles para su búsqueda en los dominios de OpenSearch Service que se ejecutan en la versión 2.13 y versiones posteriores. Los umbrales de registros lentos de solicitudes de búsqueda se configuran para el tiempo total de procesamiento de las solicitudes. Esto es diferente de los registros lentos de solicitudes de particiones, que se configuran para el tiempo individual de procesamiento de las particiones.

Puede especificar los registros lentos de solicitudes de búsqueda con la configuración del clúster. Esto difiere de los registros lentos de particiones, que se habilitan con la configuración del índice. Por ejemplo, puede especificar la siguiente configuración a través de la API de REST de OpenSearch:

```
PUT domain-endpoint/_cluster/settings
{
  "transient": {
    "cluster.search.request.slowlog.threshold.warn": "5s",
    "cluster.search.request.slowlog.threshold.info": "2s"
  }
}
```

Establecimiento de umbrales de registros lentos de particiones

OpenSearch deshabilita los registros lentos de particiones de forma predeterminada. Después de habilitar la publicación de registros lentos de particiones en CloudWatch, aún necesita especificar los umbrales de registro para cada índice de OpenSearch. Estos umbrales definen exactamente lo que se debe registrar y el nivel de registro.

Por ejemplo, puede especificar esta configuración a través de la API de REST de OpenSearch:

```
PUT domain-endpoint/index/_settings
{
  "index.search.slowlog.threshold.query.warn": "5s",
  "index.search.slowlog.threshold.query.info": "2s"
}
```

Prueba de registros lentos

Para probar que los registros lentos de particiones y solicitudes de búsqueda se publiquen correctamente, considere la posibilidad de empezar con valores muy bajos para verificar que los registros aparezcan en CloudWatch y, a continuación, aumente los umbrales a niveles más útiles.

Si los registros no aparecen, verifique lo siguiente:

- ¿Existe el grupo de registros de CloudWatch? Verifique la consola de CloudWatch.
- ¿OpenSearch Service tiene permisos para escribir en el grupo de registros? Verifique la consola de OpenSearch Service.
- ¿El dominio de OpenSearch Service está configurado para publicar en el grupo de registros? Verifique la consola de OpenSearch Service, utilice la opción AWS CLI de `describe-domain-config` o llame a `DescribeDomainConfig` mediante uno de los SDK.
- ¿Los umbrales de registro de OpenSearch son tan bajos que las solicitudes los superan?

  Para revisar los umbrales de registros lentos de solicitudes de búsqueda para un dominio, use el siguiente comando:

  ```
  GET domain-endpoint/_cluster/settings?flat_settings
  ```

  Para revisar los umbrales de registros lentos de particiones, use el siguiente comando:

  ```
  GET domain-endpoint/index/_settings?pretty
  ```

Si desea deshabilitar los registros lentos de un índice, devuelva los umbrales que ha cambiado a sus valores predeterminados de `-1`.

Deshabilitar la publicación en CloudWatch mediante la consola de OpenSearch Service o AWS CLI no impide la generación de registros por parte de OpenSearch; solo se detiene la publicación de esos registros. Asegúrese de verificar la configuración de su índice si ya no necesita registros lentos de particiones, y la configuración de su dominio si ya no necesita registros lentos de solicitudes de búsqueda.

Visualización de registros

La visualización de la aplicación y de los registros lentos en CloudWatch es exactamente igual que la visualización de cualquier otro registro de CloudWatch. Para más información, consulte View Log Data en la Guía del usuario de Registros de Amazon CloudWatch.

Aquí tiene algunas consideraciones para visualizar los registros:

- OpenSearch Service publica solo los primeros 255 000 caracteres de cada línea en CloudWatch. El contenido restante está truncado. Para los registros de auditoría, son 10 000 caracteres por mensaje.
- En CloudWatch, los nombres del flujo de registro tienen los sufijos `-index-slow-logs`, `-search-slow-logs`, `-application-logs` y `-audit-logs` para ayudar a identificar su contenido.

Monitoreo de registros de auditoría en Amazon OpenSearch Service

Si el dominio de Amazon OpenSearch Service utiliza un control de acceso detallado, puede habilitar registros de auditoría para los datos. Los registros de auditoría se pueden personalizar en gran medida y permiten realizar un seguimiento de la actividad de los usuarios en los clústeres de OpenSearch, incluidos los aciertos y los errores de autenticación, las solicitudes a OpenSearch, los cambios de índice y las consultas de búsqueda entrantes. La configuración predeterminada realiza un seguimiento de un conjunto común de acciones de usuario, pero recomendamos adaptar la configuración a las necesidades exactas.

Al igual que con los Registros de aplicaciones y registros lentos de OpenSearch, OpenSearch Service publica registros de auditoría en CloudWatch Logs. Si están habilitados, se aplica el precio estándar de CloudWatch.

> **Note**
>
> Para habilitar los registros de auditoría, el rol de usuario debe ser asignado al rol `security_manager`, que brinda acceso a la API REST `plugins/_security` de OpenSearch. Para obtener más información, consulte the section called “Modificar el usuario maestro”.

Temas

- Limitaciones
- Habilitación de los registros de auditoría
- Habilite el registro de auditoría con la AWS CLI
- Habilite el registro de auditoría con la API de configuración

- Categorías y capas de registro de auditoría
- Configuración de registros de auditoría
- Ejemplo de registro de auditoría
- Configuración de registros de auditoría mediante la API REST

Limitaciones

Los registros de auditoría presentan las siguientes limitaciones:

- Los registros de auditoría no incluyen peticiones de búsqueda entre clústeres rechazadas por la política de acceso al dominio de destino.
- El tamaño máximo de cada mensaje de registro de auditoría es de 10 000 caracteres. El mensaje del registro de auditoría se trunca si se supera este límite.

Habilitación de los registros de auditoría

La habilitación de registros de auditoría es un proceso de dos pasos. Primero, debe configurar el dominio para publicar registros de auditoría en CloudWatch Logs. Después, debe habilitar los registros de auditoría en OpenSearch Dashboards y configurarlos según sus necesidades.

> **Important**
>
> Si encuentra un error al seguir estos pasos, consulte the section called “No se pueden habilitar los registros de auditoría” para obtener información para la resolución de problemas.

Paso 1: Habilitar los registros de auditoría y configurar una política de acceso

En estos pasos se describe cómo habilitar los registros de auditoría con la consola. También puede habilitarlos mediante la AWS CLI o la API del servicio OpenSearch.

Para habilitar los registros de auditoría para un dominio de OpenSearch Service (consola)

1. Seleccione el dominio para abrir su configuración y, a continuación, vaya a la pestaña Registros.
2. Seleccione Registros de auditoría y, luego Habilitar.
3. Cree un grupo de registros en Registros de CloudWatch o elija uno existente.

4. Seleccione una política de acceso que contenga los permisos adecuados o cree una política mediante el JSON que proporciona la consola:

   JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "es.amazonaws.com"
      },
      "Action": [
        "logs:PutLogEvents",
        "logs:CreateLogStream"
      ],
      "Resource": "arn:aws:logs:us-east-1:111122223333:log-group:/aws/
opensearch/domains/domain-name/*"
    }
  ]
}
```

   Le recomendamos que agregue las claves de condición `aws:SourceAccount` y `aws:SourceArn` para protegerse contra la política problema del suplente confuso. La cuenta de origen es la propietaria del dominio y el ARN de origen es el ARN del dominio. Su dominio debe estar en el software de servicio R20211203 o posterior para agregar estas claves de condición.

   Por ejemplo, podría agregar el siguiente bloque de condición a la política:

```
"Condition": {
    "StringEquals": {
        "aws:SourceAccount": "account-id"
    },
    "ArnLike": {
        "aws:SourceArn": "arn:aws:es:region:account-id:domain/domain-name"
    }
}
```

5. Seleccione Habilitar.

Paso 2: Activar los registros de auditoría en OpenSearch Dashboards

Después de habilitar los registros de auditoría en la consola de OpenSearch Service, también debe habilitarlos en OpenSearch Dashboards y configurarlos según sus necesidades.

1. Abra OpenSearch Dashboards y seleccione Seguridad del menú a la izquierda.
2. Seleccione Registros de auditoría.
3. Seleccione Habilitar registro de auditoría.

La interfaz de usuario de Dashboards ofrece un control total de la configuración del registro de auditoría en Configuración general y en Configuración de la conformidad. Para obtener una descripción de todas las opciones de configuración, consulte Configuración del registro de auditoría.

Habilite el registro de auditoría con la AWS CLI

El siguiente comando de la AWS CLI habilita los registros de auditoría en un dominio existente:

```
aws opensearch update-domain-config --domain-name my-domain --log-publishing-options
 "AUDIT_LOGS={CloudWatchLogsLogGroupArn=arn:aws:logs:us-east-1:123456789012:log-
group:my-log-group,Enabled=true}"
```

También puede habilitar registros de auditoría al crear un dominio. Para obtener más información, consulte la Referencia de comandos de la AWS CLI.

Habilite el registro de auditoría con la API de configuración

La siguiente solicitud a la API de configuración habilita los registros de auditoría en un dominio existente:

```
POST https://es.us-east-1.amazonaws.com/2021-01-01/opensearch/domain/my-domain/config
{
  "LogPublishingOptions": {
    "AUDIT_LOGS": {
      "CloudWatchLogsLogGroupArn":"arn:aws:logs:us-east-1:123456789012:log-
group1:sample-domain",
      "Enabled":true
    }
  }
```

```
}
```

Para obtener más información, consulte la Referencia de la API de Amazon OpenSearch Service.

Categorías y capas de registro de auditoría

La comunicación del clúster se produce a través de dos capas independientes: la capa REST y la capa de transporte.

- La capa REST se encarga de la comunicación con clientes HTTP como curl, Logstash, OpenSearch Dashboards, el cliente REST de alto nivel de Java, la biblioteca de solicitudes de Python; todas las solicitudes HTTP que llegan al clúster.
- La capa de transporte se encarga de la comunicación entre nodos. Por ejemplo, después de que una petición de búsqueda llega al clúster (en la capa REST), el nodo de coordinación que se encarga de la solicitud envía la consulta a otros nodos, recibe sus respuestas, reúne los documentos necesarios y los reúne en la respuesta final. Las operaciones como la asignación de particiones y el reequilibrio también se producen en la capa de transporte.

Puede habilitar o deshabilitar los registros de auditoría para capas enteras, así como las categorías de auditoría individuales de una capa. La siguiente tabla contiene un resumen de las categorías de auditoría y las capas para las que están disponibles.

Categoría	Descripción	Disponible para REST	Disponible para transporte
FAILED_LOGIN	Una solicitud contenía credenciales no válidas y la autenticación falló.	Sí	Sí
MISSING_PRIVILEGES	Un usuario no poseía los privilegios para realizar la solicitud.	Sí	Sí
GRANTED_PRIVILEGES	Un usuario poseía los privilegios para realizar la solicitud.	Sí	Sí

Categoría	Descripción	Disponible para REST	Disponible para transporte
OPENSEARCH_SECURITY_INDEX_ATTEMPT	Una solicitud intentó modificar el índice `.opendistro_security` .	No	Sí
AUTHENTICATED	Una solicitud contenía credenciales válidas y la autenticación se realizó correctamente.	Sí	Sí
INDEX_EVENT	Una solicitud realizó una operación administrativa con un índice, como crearlo, establecer un alias o realizar una combinación de fuerza. La lista completa de acciones `indices:admin/` que incluye esta categoría están disponibles en la documentación de OpenSearch.	No	Sí

Además de estas categorías estándar, el control de acceso detallado ofrece varias categorías adicionales diseñadas para cumplir con los requisitos de conformidad de los datos.

Categoría	Descripción
COMPLIANCE_DOC_READ	Una solicitud realizó un evento de lectura en un documento en un índice.

Categoría	Descripción
COMPLIANCE_DOC_WRITE	Una solicitud realizó un evento de escritura en un documento en un índice.
COMPLIANCE_INTERNAL_CONFIG_READ	Una solicitud realizó un evento de lectura en el índice `.opendistro_security` .
COMPLIANCE_INTERNAL_CONFIG_WRITE	Una solicitud realizó un evento de escritura en el índice `.opendistro_security` .

Puede utilizar cualquier combinación de categorías y atributos de mensaje. Por ejemplo, si envía una solicitud REST para indexar un documento, podría ver las siguientes líneas en los registros de auditoría:

- AUTHENTICATED en la capa REST (autenticación)
- GRANTED_PRIVILEGE en la capa de transporte (autorización)
- COMPLIANCE_DOC_WRITE (documento escrito en un índice)

Configuración de registros de auditoría

Los registros de auditoría cuentan con numerosas opciones de configuración.

Configuración general

La configuración general permite habilitar o deshabilitar categorías individuales o capas enteras. Recomendamos encarecidamente mantener GRANTED_PRIVILEGES y AUTHENTICATED como categorías excluidas. De lo contrario, estas categorías se registran en cada solicitud válida al clúster.

Nombre	Configuración del backend	Descripción
Capa REST	enable_rest	Habilite o deshabilite los eventos que se producen en la capa REST.

Nombre	Configuración del backend	Descripción
Categorías REST deshabilitadas	disabled_rest_categories	Especifique las categorías de auditoría que se omitirán en la capa REST. La modificación de estas categorías puede aumentar drásticamente el tamaño de los registros de auditoría.
Capa de transporte	enable_transport	Habilite o deshabilite eventos que ocurren en la capa de transporte.
Categorías de transporte deshabilitadas	disabled_transport_categories	Especifique las categorías de auditoría que deben omitirse en la capa de transporte. La modificación de estas categorías puede aumentar drásticamente el tamaño de los registros de auditoría.

La configuración de atributos permite personalizar la cantidad de detalles en cada línea de registro.

Nombre	Configuración del backend	Descripción
Solicitudes masivas	resolve_bulk_requests	Habilitar esta configuración genera un registro para cada documento de una solicitud masiva, lo que puede aumentar drásticamente el tamaño de los registros de auditoría.
Cuerpo de la solicitud	log_request_body	Incluya el cuerpo de la solicitud de las solicitudes.
Resolver índices	resolve_indices	Resolver alias a índices.

Utilice la configuración de ignorar para excluir un conjunto de usuarios o rutas de API:

Nombre	Configuración del backend	Descripción
Usuarios ignorados	ignore_users	Especifique los usuarios que desea excluir.
Solicitudes ignoradas	ignore_requests	Especifique los patrones de solicitud que desea excluir.

Configuración de la conformidad

La configuración de la conformidad permite ajustar el acceso a nivel del índice, del documento o del campo.

Nombre	Configuración del backend	Descripción
Registro de conformidad	enable_compliance	Para habilitar o deshabilitar el registro de conformidad.

Puede especificar la siguiente configuración para el registro de eventos de lectura y escritura.

Nombre	Configuración del backend	Descripción
Registro de configuración interno	internal_config	Habilite o deshabilite el registro de eventos en el índice `.opendistro_security` .

Puede especificar la siguiente configuración para el registro de eventos de lectura.

Nombre	Configuración del backend	Descripción
Metadatos de lectura	read_metadata_only	Incluya solo metadatos para eventos de lectura. No incluya ningún campo de documento.
Usuarios ignorados	read_ignore_users	No incluya ciertos usuarios para eventos de lectura.
Campos observados	read_watched_fields	Especifique los índices y campos que se van a observar en busca de eventos de lectura. Agregar campos vigilados genera un registro por cada acceso a los documentos, lo que puede aumentar drásticamente el tamaño de los registros de auditoría. Los campos observados admiten patrones de índice y patrones de campo: `{ "index-name-pattern": [ "field-name-pattern" ], "logs*": [ "message" ], "twitter": [ "id", "user*" ] }`

Puede especificar la siguiente configuración para eventos de escritura.

Nombre	Configuración del backend	Descripción
Metadatos de escritura	write_metadata_only	Incluya solo metadatos para eventos de escritura. No incluya ningún campo de documento.

Nombre	Configuración del backend	Descripción
Diferencias de registro	write_log_diffs	Si write_metadata_only es falso, incluya solo las diferencias entre los eventos de escritura.
Usuarios ignorados	write_ignore_users	No incluya ciertos usuarios para eventos de escritura.
Índices de seguimiento	write_watched_indices	Especifique los índices o patrones de índice para observar los eventos de escritura. Agregar campos vigilados genera un registro por cada acceso a los documentos, lo que puede aumentar drásticamente el tamaño de los registros de auditoría.

Ejemplo de registro de auditoría

Esta sección incluye un ejemplo de configuración, una petición de búsqueda y el registro de auditoría resultante para todos los eventos de lectura y escritura de un índice.

Paso 1: configurar los registros de auditoría

Después de habilitar la publicación de registros de auditoría en un grupo de CloudWatch Logs, vaya a la página de registro de auditoría de OpenSearch Dashboards y elija Habilitar registro de auditoría.

1. En Configuración general, seleccione Configurar y asegúrese de que la capa REST esté habilitada.
2. En Configuración de la conformidad, seleccione Configurar.
3. En Escritura, en Campos observados, agregue `accounts` para todos los eventos de escritura en este índice.
4. En Lectura, en Campos observados, agregue los campos `ssn` y `id-` del índice `accounts`:

```
{
  "accounts-": [
    "ssn",
    "id-"
  ]
}
```

Paso 2: realizar eventos de lectura y escritura

1. Vaya a OpenSearch Dashboards, seleccione Herramientas para desarrolladores e indexe un documento de ejemplo:

```
PUT accounts/_doc/0
{
  "ssn": "123",
  "id-": "456"
}
```

2. Para probar un evento de lectura, envíe la siguiente solicitud:

```
GET accounts/_search
{
  "query": {
    "match_all": {}
  }
}
```

Paso 3: observar los registros

1. Abra la consola de CloudWatch en https://console.aws.amazon.com/cloudwatch/.
2. En el panel de navegación, seleccione Grupos de registro.
3. Seleccione el grupo de registros que especificó al habilitar registros de auditoría. Dentro del grupo de registros, OpenSearch Service crea un flujo de registros para cada nodo del dominio.
4. En Flujos de registros, seleccione Buscar todos.
5. Para los eventos de lectura y escritura, consulte los registros correspondientes. Puede existir una demora de 5 segundos antes de que aparezca el registro.

   Ejemplo de registro de auditoría de escritura

```
{
  "audit_compliance_operation": "CREATE",
  "audit_cluster_name": "824471164578:audit-test",
  "audit_node_name": "be217225a0b77c2bd76147d3ed3ff83c",
  "audit_category": "COMPLIANCE_DOC_WRITE",
  "audit_request_origin": "REST",
  "audit_compliance_doc_version": 1,
```

```
  "audit_node_id": "3xNJhm4XS_yTzEgDWcGRjA",
  "@timestamp": "2020-08-23T05:28:02.285+00:00",
  "audit_format_version": 4,
  "audit_request_remote_address": "3.236.145.227",
  "audit_trace_doc_id": "lxnJGXQBqZSlDB91r_uZ",
  "audit_request_effective_user": "admin",
  "audit_trace_shard_id": 8,
  "audit_trace_indices": [
    "accounts"
  ],
  "audit_trace_resolved_indices": [
    "accounts"
  ]
}
```

Ejemplo de registro de auditoría de lectura

```
{
  "audit_cluster_name": "824471164578:audit-docs",
  "audit_node_name": "806f6050cb45437e2401b07534a1452f",
  "audit_category": "COMPLIANCE_DOC_READ",
  "audit_request_origin": "REST",
  "audit_node_id": "saSevm9ASte0-pjAtYi2UA",
  "@timestamp": "2020-08-31T17:57:05.015+00:00",
  "audit_format_version": 4,
  "audit_request_remote_address": "54.240.197.228",
  "audit_trace_doc_id": "config:7.7.0",
  "audit_request_effective_user": "admin",
  "audit_trace_shard_id": 0,
  "audit_trace_indices": [
    "accounts"
  ],
  "audit_trace_resolved_indices": [
    "accounts"
  ]
}
```

Para incluir el cuerpo de la solicitud, vuelva a Configuración de la conformidad en OpenSearch Dashboards y deshabilite Metadatos de escritura. Para excluir eventos de un usuario específico, agregue el usuario a Usuarios ignorados.

Para obtener una descripción de cada campo de registro de auditoría, consulte la Referencia de campo del registro de auditoría. Para obtener información sobre cómo buscar y analizar los datos del registro de auditoría, consulte Análisis de los datos de registros con Información de registros de Amazon CloudWatch en la Guía del usuario de Registros de Amazon CloudWatch.

Configuración de registros de auditoría mediante la API REST

Recomendamos utilizar OpenSearch Dashboards para configurar los registros de auditoría, pero también puede utilizar la API REST de control de acceso detallado. Esta sección contiene una solicitud de ejemplo. La documentación completa sobre la API de REST está disponible en la documentación de OpenSearch.

```
PUT _opendistro/_security/api/audit/config
{
  "enabled": true,
  "audit": {
    "enable_rest": true,
    "disabled_rest_categories": [
      "GRANTED_PRIVILEGES",
      "AUTHENTICATED"
    ],
    "enable_transport": true,
    "disabled_transport_categories": [
      "GRANTED_PRIVILEGES",
      "AUTHENTICATED"
    ],
    "resolve_bulk_requests": true,
    "log_request_body": true,
    "resolve_indices": true,
    "exclude_sensitive_headers": true,
    "ignore_users": [
      "kibanaserver"
    ],
    "ignore_requests": [
      "SearchRequest",
      "indices:data/read/*",
      "/_cluster/health"
    ]
  },
  "compliance": {
    "enabled": true,
    "internal_config": true,
```

```
    "external_config": false,
    "read_metadata_only": true,
    "read_watched_fields": {
      "read-index-1": [
        "field-1",
        "field-2"
      ],
      "read-index-2": [
        "field-3"
      ]
    },
    "read_ignore_users": [
      "read-ignore-1"
    ],
    "write_metadata_only": true,
    "write_log_diffs": false,
    "write_watched_indices": [
      "write-index-1",
      "write-index-2",
      "log-*",
      "*"
    ],
    "write_ignore_users": [
      "write-ignore-1"
    ]
  }
}
```

Monitoreo de eventos de OpenSearch Service con Amazon EventBridge

Amazon OpenSearch Service se integra con Amazon EventBridge para informar sobre determinados eventos que afectan a los dominios. Los eventos de los servicios de AWS se envían a EventBridge casi en tiempo real. Los mismos eventos también se envían a Eventos de Amazon CloudWatch, el predecesor de Amazon EventBridge. Puede crear reglas sencillas para indicar qué eventos le resultan de interés, así como qué acciones automatizadas se van a realizar cuando un evento cumple una de las reglas. Entre las acciones que se pueden activar automáticamente se incluyen las siguientes:

- Invocar una función de AWS Lambda

- Invocar un Run Command de Amazon EC2
- Desviar el evento a Amazon Kinesis Data Streams
- Activación de una máquina de estado de AWS Step Functions
- Notificar un tema de Amazon SNS o una cola de Amazon SQS

Para más información, consulte Introducción a Amazon EventBridge en la Guía del usuario de Amazon EventBridge.

Temas

- Eventos de actualización del software de servicio
- Eventos de ajuste automático
- Eventos del estado del clúster
- Eventos del punto de conexión de VPC
- Eventos de retirada de nodos
- Eventos de retirada de nodos degradados
- Eventos de error de dominio
- Tutorial: Listening for Amazon OpenSearch Service EventBridge events
- Tutorial: Sending Amazon SNS alerts for available software updates

Eventos de actualización del software de servicio

OpenSearch Service envía eventos a EventBridge cuando se produce uno de los siguientes eventos de actualización del software de servicio.

Actualización del software de servicio disponible

OpenSearch Service envía este evento cuando hay disponible una actualización del software de servicio.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
  "version": "0",
```

```
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Software Update Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Service Software Update",
    "status": "Available",
    "severity": "Informational",
    "description": "Service software update R20220928 available. Service Software
 Deployment Mechanism:
                   Blue/Green. For more information on deployment configuration,
 please
                   see: https://docs.aws.amazon.com/opensearch-service/latest/
developerguide/managedomains-configuration-changes.html"
  }
}
```

Actualización del software de servicio programada

OpenSearch Service envía este evento cuando se programa una actualización del software de servicio. En el caso de las actualizaciones opcionales, recibirá la notificación en la fecha programada y tendrá la opción de volver a programarla en cualquier momento. En el caso de las actualizaciones obligatorias, recibirá la notificación tres días antes de la fecha programada y tendrá la opción de reprogramarla dentro del plazo obligatorio.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Software Update Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Service Software Update",
```

```
    "status": "Scheduled",
    "severity": "High",
    "description": "A new service software update [R20200330-p1] has been scheduled at
 [21st May 2023 12:40 GMT].
                    Please see documentation for more information on scheduling
 software updates:
                    https://docs.aws.amazon.com/opensearch-service/latest/
developerguide/service-software.html."
  }
}
```

Actualización del software de servicio reprogramada

OpenSearch Service envía este evento cuando se reprograma una actualización opcional del software de servicio. Para más información, consulte the section called “Actualizaciones opcionales frente a actualizaciones obligatorias”.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Software Update Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Service Software Update",
    "status": "Rescheduled",
    "severity": "High",
    "description": "The service software update [R20200330-p1], which was originally
 scheduled for
                    [21st May 2023 12:40 GMT], has been rescheduled to [23rd May 2023
 12:40 GMT].
                    Please see documentation for more information on scheduling
 software updates:
                    https://docs.aws.amazon.com/opensearch-service/latest/
developerguide/service-software.html."
  }
```

```
}
```

Actualización del software de servicio comenzada

OpenSearch Service envía este evento cuando se inicia una actualización del software de servicio.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Software Update Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Service Software Update",
    "status": "Started",
    "severity": "Informational",
    "description": "Service software update [R20200330-p1] started.
  }
}
```

Actualización del software de servicio finalizada

OpenSearch Service envía este evento cuando se completó una actualización del software de servicio.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Software Update Notification",
  "source": "aws.es",
  "account": "123456789012",
```

```
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Service Software Update",
    "status": "Completed",
    "severity": "Informational",
    "description": "Service software update [R20200330-p1] completed."
  }
}
```

Actualización del software de servicio cancelada

OpenSearch Service envía este evento cuando se cancela una actualización del software de servicio.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Software Update Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Service Software Update",
    "status": "Cancelled",
    "severity": "Informational",
    "description": "The scheduled service software update [R20200330-p1] has been
 cancelled as a
                    newer update is available. Please schedule the latest update."
  }
}
```

Actualización programada del software de servicio cancelada

OpenSearch Service envía este evento cuando se cancela una actualización del software de servicio que estaba programada previamente para el dominio.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Software Update Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Service Software Update",
    "status": "Cancelled",
    "severity": "Informational",
    "description": "The scheduled service software update [R20200330-p1] has been
 cancelled."
  }
}
```

Actualización del software de servicio no ejecutada

OpenSearch Service envía este evento cuando no puede iniciar una actualización del software de servicio.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Software Update Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Service Software Update",
    "status": "Unexecuted",
```

```
    "severity": "Informational",
    "description": "The scheduled service software update [R20200330-p1] cannot be
 started. Reason: [reason]"
  }
}
```

Error al actualizar el software del servicio

OpenSearch Service envía este evento cuando se produce un error al actualizar el software del servicio.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Software Update Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Service Software Update",
    "status": "Failed",
    "severity": "High",
    "description": "Installation of service software update [R20200330-p1] failed.
 [reason].
  }
}
```

Se requiere actualización del software de servicio

OpenSearch Service envía este evento cuando se requiere una actualización del software de servicio. Para más información, consulte the section called “Actualizaciones opcionales frente a actualizaciones obligatorias”.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Software Update Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Service Software Update",
    "status": "Required",
    "severity": "High",
    "description": "Service software update [R20200330-p1] available. Update
                    will be automatically installed after [21st May 2023] if no
                    action is taken. Service Software Deployment Mechanism: Blue/Green.
                    For more information on deployment configuration, please see:
                    https://docs.aws.amazon.com/opensearch-service/latest/
developerguide/managedomains-configuration-changes.html"
  }
}
```

Eventos de ajuste automático

OpenSearch Service envía eventos a EventBridge cuando se produce uno de los siguientes eventos de ajuste automático.

Ajuste automático pendiente

OpenSearch Service envía este evento cuando el ajuste automático identifica recomendaciones de ajuste para mejorar el rendimiento y la disponibilidad del clúster. Solo verá este evento para dominios con el ajuste automático deshabilitado.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
  "version": "0",
  "id": "3acb26c8-397c-4c89-a80a-ce672a864c55",
  "detail-type": "Amazon OpenSearch Service Auto-Tune Notification",
```

```
  "source": "aws.es",
  "account": "123456789012",
  "time": "2020-10-30T22:06:31Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Auto-Tune Event",
    "severity": "Informational",
    "status": "Pending",
    "description": "Auto-Tune recommends the following new settings for your
 domain: { JVM Heap size : 60%}. Enable Auto-Tune to improve cluster stability and
 performance.",
    "scheduleTime": "{iso8601-timestamp}"
  }
}
```

Ajuste automático iniciado

OpenSearch Service envía este evento cuando el ajuste automático comienza a aplicar una nueva configuración al dominio.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
  "version": "0",
  "id": "3acb26c8-397c-4c89-a80a-ce672a864c55",
  "detail-type": "Amazon OpenSearch Service Auto-Tune Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2020-10-30T22:06:31Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Auto-Tune Event",
    "severity": "Informational",
    "status": "Started",
    "scheduleTime": "{iso8601-timestamp}",
    "startTime": "{iso8601-timestamp}",
    "description" : "Auto-Tune is applying the following settings to your domain: { JVM
 Heap size : 60%}."
  }
```

```
}
```

El ajuste automático requiere una implementación azul/verde programada

OpenSearch Service envía este evento cuando el ajuste automático identifica recomendaciones de ajuste que requieren una implementación azul/verde programada.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
  "version": "0",
  "id": "3acb26c8-397c-4c89-a80a-ce672a864c55",
  "detail-type": "Amazon OpenSearch Service Auto-Tune Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2020-10-30T22:06:31Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Auto-Tune Event",
    "severity": "Low",
    "status": "Pending",
    "startTime": "{iso8601-timestamp}",
    "description": "Auto-Tune has identified the following settings for your domain
 that require a blue/green deployment: { JVM Heap size : 60%}.
                   You can schedule the deployment for your preferred time."
  }
}
```

Ajuste automático cancelado

OpenSearch Service envía este evento cuando se cancela la programación de ajuste automático porque no hay recomendaciones de ajuste pendientes.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
  "version": "0",
```

```
  "id": "3acb26c8-397c-4c89-a80a-ce672a864c55",
  "detail-type": "Amazon OpenSearch Service Auto-Tune Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2020-10-30T22:06:31Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Auto-Tune Event",
    "severity": "Low",
    "status": "Cancelled",
    "scheduleTime": "{iso8601-timestamp}",
    "description": "Auto-Tune has cancelled the upcoming blue/green deployment."
  }
}
```

Ajuste automático completado

OpenSearch Service envía este evento cuando el ajuste automático completa la implementación azul/verde y el clúster está en funcionamiento con la nueva configuración de JVM instalada.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
  "version": "0",
  "id": "3acb26c8-397c-4c89-a80a-ce672a864c55",
  "detail-type": "Amazon OpenSearch Service Auto-Tune Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2020-10-30T22:06:31Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Auto-Tune Event",
    "severity": "Informational",
    "status": "Completed",
    "completionTime": "{iso8601-timestamp}",
    "description": "Auto-Tune has completed the blue/green deployment and successfully
 applied the following settings: { JVM Heap size : 60%}."
  }
}
```

Ajuste automático deshabilitado y cambios revertidos

OpenSearch Service envía este evento cuando se deshabilita el ajuste automático y se revierten los cambios aplicados.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
  "version": "0",
  "id": "3acb26c8-397c-4c89-a80a-ce672a864c55",
  "detail-type": "Amazon OpenSearch Service Auto-Tune Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2020-10-30T22:06:31Z",
  "region": "us-east-1",
  "resources": [ "arn:aws:es:us-east-1:123456789012:domain/test-domain" ],
  "detail": {
    "event": "Auto-Tune Event",
    "severity": "Informational",
    "status": "Completed",
    "description": "Auto-Tune is now disabled. All settings have been reverted. Auto-
Tune will continue to evaluate
                    cluster performance and provide recommendations.",
    "completionTime": "{iso8601-timestamp}"
  }
}
```

Ajuste automático deshabilitado y cambios conservados

OpenSearch Service envía este evento cuando se deshabilita el ajuste automático y se conservan los cambios aplicados.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
  "version": "0",
  "id": "3acb26c8-397c-4c89-a80a-ce672a864c55",
  "detail-type": "Amazon OpenSearch Service Auto-Tune Notification",
  "source": "aws.es",
```

```
  "account": "123456789012",
  "time": "2020-10-30T22:06:31Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Auto-Tune Event",
    "severity": "Informational",
    "status": "Completed",
    "description": "Auto-Tune is now disabled. The most-recent settings by Auto-Tune
 have been retained.
                    Auto-Tune will continue to evaluate cluster performance and provide
 recommendations.",
    "completionTime": "{iso8601-timestamp}"
  }
}
```

Eventos del estado del clúster

OpenSearch Service envía ciertos eventos a EventBridge cuando el estado del clúster se ve comprometido.

Red cluster recovery started (Recuperación de clúster en rojo iniciada)

OpenSearch Service envía este evento cuando el estado del clúster ha permanecido en rojo de manera continuada durante más de una hora. Intenta restaurar automáticamente uno o varios índices en rojo a partir de una instantánea para corregir el estado del clúster.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-012345678901",
   "detail-type":"Amazon OpenSearch Service Cluster Status Notification",
   "source":"aws.es",
   "account":"123456789012",
   "time":"2016-11-01T13:12:22Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-domain"
   ],
   "detail":{
```

```
      "event":"Automatic Snapshot Restore for Red Indices",
      "status":"Started",
      "severity":"High",
      "description":"Your cluster status is red. We have started automatic snapshot
 restore for the red indices.
                    No action is needed from your side. Red indices [red-index-0, red-
index-1]"
   }
}
```

Red cluster recovery partially completed (Recuperación de clúster en rojo completada parcialmente)

OpenSearch Service envía este evento cuando solo ha podido restaurar un subconjunto de índices en rojo a partir de una instantánea mientras intentaba corregir el estado de un clúster en rojo.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-012345678901",
   "detail-type":"Amazon OpenSearch Service Cluster Status Notification",
   "source":"aws.es",
   "account":"123456789012",
   "time":"2016-11-01T13:12:22Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-domain"
   ],
   "detail":{
      "event":"Automatic Snapshot Restore for Red Indices",
      "status":"Partially Restored",
      "severity":"High",
      "description":"Your cluster status is red. We were able to restore the following
 Red indices from
                   snapshot: [red-index-0]. Indices not restored: [red-index-1].
 Please refer https://docs.aws.amazon.com/opensearch-service/latest/developerguide/
handling-errors.html#handling-errors-red-cluster-status for troubleshooting steps."
   }
}
```

Red cluster recovery failed (Error de recuperación de clúster en rojo)

OpenSearch Service envía este evento cuando no puede restaurar ningún índice mientras intenta corregir el estado de un clúster en rojo.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-012345678901",
   "detail-type":"Amazon OpenSearch Service Cluster Status Notification",
   "source":"aws.es",
   "account":"123456789012",
   "time":"2016-11-01T13:12:22Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-domain"
   ],
   "detail":{
      "event":"Automatic Snapshot Restore for Red Indices",
      "status":"Failed",
      "severity":"High",
      "description":"Your cluster status is red. We were unable to restore the Red
 indices automatically.
                    Indices not restored: [red-index-0, red-index-1]. Please refer
 https://docs.aws.amazon.com/opensearch-service/latest/developerguide/handling-
errors.html#handling-errors-red-cluster-status for troubleshooting steps."
   }
}
```

Shards to be deleted (Se deben eliminar particiones)

OpenSearch Service envía este evento cuando ha intentado corregir automáticamente el estado del clúster en rojo después de que haya permanecido en rojo de manera continuada durante 14 días, pero uno o varios índices permanecen en rojo. Después de 7 días más (21 días en total permaneciendo en rojo de manera continuada), OpenSearch Service procede a eliminar las particiones no asignadas de todos los índices en rojo.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-012345678901",
   "detail-type":"Amazon OpenSearch Service Cluster Status Notification",
   "source":"aws.es",
   "account":"123456789012",
   "time":"2022-04-09T10:36:48Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-domain"
   ],
   "detail":{
      "severity":"Medium",
      "description":"Your cluster status is red. Please fix the red indices as soon as
 possible.
                    If not fixed by 2022-04-12 01:51:47+00:00, we will delete all
 unassigned shards,
                    the unit of storage and compute, for these red indices to recover
 your domain and make it green.
                    Please refer to https://docs.aws.amazon.com/opensearch-service/
latest/developerguide/handling-errors.html#handling-errors-red-cluster-status for
 troubleshooting steps.
                    test_data, test_data1",
      "event":"Automatic Snapshot Restore for Red Indices",
      "status":"Shard(s) to be deleted"
   }
}
```

Shards deleted (Particiones eliminadas)

OpenSearch Service envía este evento cuando el estado del clúster ha permanecido en rojo de manera continuada durante 21 días. Procede a eliminar las particiones no asignadas (almacenamiento y computación) de todos los índices en rojo. Para más información, consulte the section called “Corrección automática de clústeres en rojo”.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-012345678901",
   "detail-type":"Amazon OpenSearch Service Cluster Status Notification",
```

```
   "source":"aws.es",
   "account":"123456789012",
   "time":"2022-04-09T10:54:48Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-domain"
   ],
   "detail":{
      "severity":"High",
      "description":"We have deleted unassinged shards, the unit of storage and
 compute, in
                    red indices: index-1, index-2 because these indices were red for
 more than
                    21 days and could not be restored with the automated restore
 process.
                    Please refer to https://docs.aws.amazon.com/opensearch-service/
latest/developerguide/handling-errors.html#handling-errors-red-cluster-status for
 troubleshooting steps.",
      "event":"Automatic Snapshot Restore for Red Indices",
      "status":"Shard(s) deleted"
   }
}
```

Advertencia de alto recuento de particiones

OpenSearch Service envía este evento cuando el recuento medio de particiones de los nodos de datos calientes ha superado el 90 % del límite predeterminado recomendado de 1000. Aunque las versiones posteriores de Elasticsearch y OpenSearch soportan un recuento máximo de particiones configurable por nodo, le recomendamos que no tenga más de 1000 particiones por nodo. Consulte Elección del número de particiones.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2016-11-01T13:12:22Z",
  "region":"us-east-1",
```

```
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"High Shard Count",
     "status":"Warning",
     "severity":"Low",
     "description":"One or more data nodes have close to 1000 shards. To ensure optimum
 performance and stability of your
                   cluster, please refer to the best practice guidelines - https://
docs.aws.amazon.com/opensearch-service/latest/developerguide/sizing-domains.html#bp-
sharding."
  }
}
```

Límite de recuento de particiones superado

OpenSearch Service envía este evento cuando el recuento medio de particiones en sus nodos de datos calientes ha superado el límite predeterminado recomendado de 1000. Aunque las versiones posteriores de Elasticsearch y OpenSearch soportan un recuento máximo de particiones configurable por nodo, le recomendamos que no tenga más de 1000 particiones por nodo. Consulte Elección del número de particiones.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2016-11-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"High Shard Count",
     "status":"Warning",
     "severity":"Medium",
     "description":"One or more data nodes have more than 1000 shards. To ensure
 optimum performance and stability of your
                   cluster, please refer to the best practice guidelines - https://
docs.aws.amazon.com/opensearch-service/latest/developerguide/sizing-domains.html#bp-
sharding."
```

```
  }
}
```

Espacio en disco reducido

OpenSearch Service envía este evento cuando uno o más nodos de su cluster tienen menos del 25 % del espacio de almacenamiento disponible, o menos de 25 GB.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2017-12-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"Low Disk Space",
     "status":"Warning",
     "severity":"Medium",
     "description":"One or more data nodes in your cluster has less than 25% of storage
 space or less than 25GB.
                  Your cluster will be blocked for writes at 20% or 20GB. Please refer
 to the documentation for more information - https://docs.aws.amazon.com/opensearch-
service/latest/developerguide/handling-errors.html#troubleshooting-cluster-block"
  }
}
```

Vulneración del límite de espacio en disco mínimo

OpenSearch Service envía este evento cuando todos los nodos del clúster tienen menos del 10 % del espacio de almacenamiento disponible, o menos de 10 GB. Si todos los nodos superan el límite de disco bajo, cualquier índice nuevo da como resultado un clúster amarillo, y si todos los nodos se sitúan por debajo del límite de disco alto, aparece un clúster rojo.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2017-12-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"Low Disk Watermark Breach",
     "status":"Warning",
     "severity":"Medium",
     "description":"Low Disk Watermark threshold is about to be breached. Once the
 threshold is breached, new index creation will be blocked on all
                   nodes to prevent the cluster status from turning red. Please
 increase disk size to suit your storage needs. For more information,
                   see https://docs.aws.amazon.com/opensearch-service/latest/
developerguide/handling-errors.html#troubleshooting-cluster-block".
  }
}
```

Balance de ráfaga de EBS por debajo del 70 %

OpenSearch Service envía este evento cuando el balance de ráfaga de EBS en uno o más nodos de datos cae por debajo del 70 %. El agotamiento del balance de ráfaga de EBS puede provocar una falta de disponibilidad generalizada del clúster y una limitación de las solicitudes de E/S, lo que puede provocar altas latencias y tiempos de espera en las solicitudes de indexación y búsqueda. Para conocer los pasos para solucionar este problema, consulte the section called “Bajo balance de ráfaga EBS”.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
```

```
  "account":"123456789012",
  "time":"2017-12-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"EBS Burst Balance",
     "status":"Warning",
     "severity":"Medium",
     "description":"EBS burst balance on one or more data nodes is below 70%.
                    Follow https://docs.aws.amazon.com/opensearch-service/latest/
developerguide/handling-errors.html#handling-errors-low-ebs-burst
                    to fix this issue."
  }
}
```

Balance de ráfaga de EBS por debajo del 20 %

OpenSearch Service envía este evento cuando el balance de ráfaga de EBS en uno o más nodos de datos cae por debajo del 20 %. El agotamiento del balance de ráfaga de EBS puede provocar una falta de disponibilidad generalizada del clúster y una limitación de las solicitudes de E/S, lo que puede provocar altas latencias y tiempos de espera en las solicitudes de indexación y búsqueda. Para conocer los pasos para solucionar este problema, consulte the section called “Bajo balance de ráfaga EBS”.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2017-12-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"EBS Burst Balance",
     "status":"Warning",
     "severity":"High",
     "description":"EBS burst balance on one or more data nodes is below 20%.
```

```
                    Follow https://docs.aws.amazon.com/opensearch-service/latest/
developerguide/handling-errors.html#handling-errors-low-ebs-burst
                    to fix this issue.
  }
}
```

Limitación de rendimiento de disco

OpenSearch Service envía este evento cuando las solicitudes de lectura y escritura en su dominio están siendo limitadas debido a las limitaciones de rendimiento de los volúmenes de EBS o de la instancia EC2. Si recibe esta notificación, considere la posibilidad de escalar verticalmente los volúmenes o las instancias siguiendo las prácticas recomendadas de AWS. Si su tipo de volumen es gp2, aumente el tamaño del volumen. Si su tipo de volumen es gp3, aprovisione más rendimiento. También puede comprobar que la base de instancias y el rendimiento máximo de EBS sean superiores o iguales al rendimiento del volumen aprovisionado, y puede escalar verticalmente en consecuencia.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2017-12-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"Disk Throughput Throttle",
     "status":"Warning",
     "severity":"Medium",
     "description":"Your domain is experiencing throttling due to instance or volume
 throughput limitations.
                    Please consider scaling your domain to suit your throughput needs.
 In July 2023, we improved
                    the accuracy of throughput throttle calculation by replacing ‘Max
 volume throughput’ with
                    ‘Provisioned volume throughput’. Please refer to the documentation
 for more information."
```

```
  }
}
```

Tamaño de partición grande

OpenSearch Service envía este evento cuando una o más particiones del clúster superan los 50 GiB o 65 GiB. Para garantizar un rendimiento y una estabilidad óptimos del clúster, reduzca el tamaño de las particiones.

Para obtener más información, consulte las prácticas recomendadas de particiones.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2017-12-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"Large Shard Size",
     "status":"Warning",
     "severity":"Medium",
     "description":"One or more shards are larger than 65GiB. To ensure optimum cluster
 performance and stability, reduce shard sizes.
                    For more information, see https://docs.aws.amazon.com/opensearch-
service/latest/developerguide/monitoring-events.html#monitoring-events-large-shard-
size."
  }
}
```

Alto uso de JVM

OpenSearch Service envía este evento cuando la métrica `JVMMemoryPressure` de su dominio supera el 80 %. Si supera el 92 % durante 30 minutos, se bloquearán todas las operaciones de escritura en el clúster. Para garantizar una estabilidad óptima del clúster, reduzca el tráfico al clúster o escale su dominio para proporcionar memoria suficiente para su carga de trabajo.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2017-12-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"High JVM Usage",
     "status":"Warning",
     "severity":"High",
     "description":"JVM memory pressure has exceeded 80%. If it exceeds 92% for 30
 minutes, all write operations to your cluster
                    will be blocked. To ensure optimum cluster stability, reduce
 traffic to the cluster or use larger instance types.
                    For more information, see https://docs.aws.amazon.com/opensearch-
service/latest/developerguide/monitoring-events.html#monitoring-events-high-jvm."
  }
}
```

GC insuficiente

El servicio OpenSearch envía este evento cuando la JVM máxima es superior al 70 % y la diferencia entre el máximo y el mínimo es inferior al 30 %. Esto puede indicar que la JVM no puede recuperar suficiente memoria para la carga de trabajo durante los ciclos de recopilación de elementos no utilizados. Esto puede provocar respuestas cada vez más lentas y latencias más altas y, en algunos casos, incluso la caída de nodos debido a que se ha agotado el tiempo de espera de las comprobaciones de estado. Para garantizar una estabilidad óptima del clúster, reduzca el tráfico al clúster o escale su dominio para proporcionar memoria suficiente para su carga de trabajo.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
  "version":"0",
```

```
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2017-12-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"Insufficient GC",
     "status":"Warning",
     "severity":"Medium",
     "description":"Maximum JVM is above 70% and JVM range is less than 30%. This may
 indicate insufficient garbage collection for your workload.
                   For more information, see https://docs.aws.amazon.com/opensearch-
service/latest/developerguide/monitoring-events.html#monitoring-events-insufficient-
gc."
  }
}
```

Advertencia de enrutamiento indexado personalizada

OpenSearch Service envía este evento cuando el dominio se encuentra en estado de procesamiento y contiene índices con una configuración personalizada de index.routing.location, lo que puede provocar que las implementaciones azul-verde se bloqueen. Compruebe que la configuración se aplique correctamente.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2017-12-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"Custom Index Routing Warning",
     "status":"Warning",
     "severity":"Medium",
```

```
     "description":"Your domain is in processing state and contains indice(s) with
 custom index.routing.allocation
                    settings which can cause blue-green deployments to get stuck.
 Verify settings are applied properly.
                    For more information, see https://docs.aws.amazon.com/opensearch-
service/latest/developerguide/monitoring-events.html#monitoring-events-index-routing."
  }
}
```

Fallo en el bloqueo de partición

OpenSearch Service envía este evento cuando el dominio está en mal estado debido a particiones no asignadas con `[ShardLockObtainFailedException]`. Para más información, consulte How do I resolve the in-memory shard lock exception in Amazon OpenSearch Service?

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2017-12-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"Failed Shard Lock",
     "status":"Warning",
     "severity":"Medium",
     "description":"Your domain is unhealthy due to unassigned shards with
 [ShardLockObtainFailedException]. For more information,
                    see https://docs.aws.amazon.com/opensearch-service/latest/
developerguide/monitoring-events.html#monitoring-events-failed-shard-lock."
}
```

Eventos del punto de conexión de VPC

OpenSearch Service envía ciertos eventos a EventBridge en relación con los puntos de conexión de la interfaz de AWS PrivateLink.

Error al crear el punto de conexión de VPC

OpenSearch Service envía este evento cuando no puede crear un punto de conexión de VPC solicitado. Este error puede ocurrir porque alcanzó el límite de la cantidad de puntos de conexión de VPC permitidos dentro de una región. También verá este error si no existe una subred ni un grupo de seguridad especificados.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-012345678901",
   "detail-type":"Amazon OpenSearch Service VPC Endpoint Notification",
   "source":"aws.es",
   "account":"123456789012",
   "time":"2016-11-01T13:12:22Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-domain"
   ],
   "detail":{
      "event":"VPC Endpoint Create Validation",
      "status":"Failed",
      "severity":"High",
      "description":"Unable to create VPC endpoint aos-0d4c74c0342343 for domain
                     arn:aws:es:eu-south-1:123456789012:domain/my-domain due to the
 following validation failures: You've reached the limit on the
                     number of VPC endpoints that you can create in the AWS Region."
   }
}
```

Error al actualizar el punto de conexión de VPC

OpenSearch Service envía este evento cuando no puede eliminar un punto de conexión de VPC solicitado.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-012345678901",
   "detail-type":"Amazon OpenSearch Service VPC Endpoint Notification",
   "source":"aws.es",
   "account":"123456789012",
   "time":"2016-11-01T13:12:22Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-domain"
   ],
   "detail":{
      "event":"VPC Endpoint Update Validation",
      "status":"Failed",
      "severity":"High",
      "description":"Unable to update VPC endpoint aos-0d4c74c0342343 for domain
                    arn:aws:es:eu-south-1:123456789012:domain/my-domain due to the
 following validation failures: <failure message>."
    }
}
```

Error al eliminar el punto de conexión de VPC

OpenSearch Service envía este evento cuando no puede eliminar un punto de conexión de VPC solicitado.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-012345678901",
   "detail-type":"Amazon OpenSearch Service VPC Endpoint Notification",
   "source":"aws.es",
   "account":"123456789012",
   "time":"2016-11-01T13:12:22Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-domain"
   ],
   "detail":{
      "event":"VPC Endpoint Delete Validation",
```

```
      "status":"Failed",
      "severity":"High",
      "description":"Unable to delete VPC endpoint aos-0d4c74c0342343 for domain
                    arn:aws:es:eu-south-1:123456789012:domain/my-domain due to the
 following validation failures: Specified subnet doesn't exist."
   }
}
```

Eventos de retirada de nodos

OpenSearch Service envía los eventos a EventBridge cuando se produce uno de los siguientes eventos de retirada de nodos.

Retirada de nodos programada

OpenSearch Service envía este evento cuando se programó la retirada de un nodo.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2023-04-07T10:07:33Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Node Retirement Notification",
    "status": "Scheduled",
    "severity": "Medium",
    "description": "An automated action to retire and replace a node has been scheduled
 on your domain.
                    The node will be replaced in the next off-peak window. For more
 information, see
                    https://docs.aws.amazon.com/opensearch-service/latest/
developerguide/monitoring-events.html."
  }
}
```

Se completó la retirada del nodo

OpenSearch Service envía este evento cuando se completa la retirada de un nodo.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2023-04-07T10:07:33Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Node Retirement Notification",
    "status": "Completed",
    "severity": "Medium",
    "description": "The node has been retired and replaced with a new node."
  }
}
```

Error al retirar el nodo

OpenSearch Service envía este evento cuando se produce un error al retirar un nodo.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2023-04-07T10:07:33Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
```

```
    "event": "Node Retirement Notification",
    "status": "Failed",
    "severity": "Medium",
    "description": "Node retirement failed. No actions are required from your end. We
 will automatically
                   retry replacing the node."
  }
}
```

Eventos de retirada de nodos degradados

OpenSearch Service envía estos eventos cuando es necesario reemplazar un nodo debido a la degradación del hardware de un nodo.

Notificación de retirada de nodos degradados

OpenSearch Service envía este evento cuando se ha programado la acción automatizada para retirar y reemplazar un nodo degradado para el dominio.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
   "version":"0",
   "id":"db233454-aad1-7676-3b15-10a84b052baa",
   "detail-type":"Amazon OpenSearch Service Notification",
   "source":"aws.es",
   "account":"123456789012",
   "time":"2024-01-11T08:16:06Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-node-replacement"
   ],
   "detail":{
      "severity":"Medium",
      "description":"An automated action to retire and replace a node has
 been scheduled on your domain. For more information, please see https://
docs.aws.amazon.com/opensearch-service/latest/developerguide/monitoring-events.html.",
      "event":"Degraded Node Retirement Notification",
      "status":"Scheduled"
   }
}
```

Finalización de la retirada de nodos degradados

OpenSearch Service envía este evento cuando se ha retirado y reemplazado un nodo degradado por uno nuevo.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
   "version":"0",
   "id":"7444215c-90f9-a52d-bcda-e85973a9a762",
   "detail-type":"Amazon OpenSearch Service Notification",
   "source":"aws.es",
   "account":"123456789012",
   "time":"2024-01-11T10:20:30Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-node-replacement"
   ],
   "detail":{
      "severity":"Medium",
      "description":"The node has been retired and replaced with a new node.",
      "event":"Degraded Node Retirement Notification",
      "status":"Completed"
   }
}
```

Errores de retirada de nodos degradados

OpenSearch Service envía este evento si se produce un error al retirar un nodo degradado.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
   "version":"0",
   "id":"c328e9bb-93b9-c0b2-b17a-df527fdf96b6",
   "detail-type":"Amazon OpenSearch Service Notification",
   "source":"aws.es",
   "account":"123456789012",
   "time":"2024-01-11T08:31:38Z",
```

```
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-node-replacement"
   ],
   "detail":{
      "severity":"Medium",
      "description":"Node retirement failed. No actions are required from your end. We
 will automatically re-try replacing the node.",
      "event":"Degraded Node Retirement Notification",
      "status":"Failed"
   }
}
```

Eventos de error de dominio

OpenSearch Service envía eventos a EventBridge cuando se produce uno de los siguientes errores de dominio.

Error en la validación de la actualización del dominio

OpenSearch Service envía este evento si encuentra uno o más errores de validación al intentar actualizar o hacer un cambio de configuración en un dominio. Para conocer los pasos para solucionar estos errores, consulte the section called “Solución de errores de validación”.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-012345678901",
   "detail-type":"Amazon OpenSearch Service Domain Update Notification",
   "source":"aws.es",
   "account":"123456789012",
   "time":"2016-11-01T13:12:22Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-domain"
   ],
   "detail":{
      "event":"Domain Update Validation",
      "status":"Failed",
```

```
      "severity":"High",
      "description":"Unable to perform updates to your domain due to the following
 validation failures: <failures>
                    Please see the documentation for more information https://
docs.aws.amazon.com/opensearch-service/latest/developerguide/managedomains-
configuration-changes.html#validation"
   }
}
```

Clave KMS inaccesible

OpenSearch Service envía este evento cuando no puede acceder a su clave AWS KMS.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Domain Error Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2016-11-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"KMS Key Inaccessible",
     "status":"Error",
     "severity":"High",
     "description":"The KMS key associated with this domain is inaccessible. You are at
 risk of losing access to your domain.
                    For more information, please refer to https://docs.aws.amazon.com/
opensearch-service/latest/developerguide/encryption-at-rest.html#disabled-key."
  }
}
```

Aislamiento de dominios

OpenSearch Service envía este evento cuando el dominio queda aislado y no puede recibir, leer ni escribir solicitudes porque la red no puede acceder a él.

Ejemplo

A continuación se muestra un ejemplo de este evento:

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2023-11-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"Domain Isolation Notification",
     "status":"Error",
     "severity":"High",
     "description":"Your OpenSearch Service domain has been isolated. An isolated
 domain is unreachable by network and cannot receive, read, or write requests. For more
 information and assistance, please contact AWS Support at https://docs.aws.amazon.com/
opensearch-service/latest/developerguide/encryption-at-rest.html#disabled-key."
  }
}
```

Tutorial: Listening for Amazon OpenSearch Service EventBridge events

En este tutorial, configurará una función AWS Lambda sencilla que escucha eventos de Amazon OpenSearch Service y los escribe en un flujo de registros en Registros de CloudWatch.

Requisitos previos

En este tutorial se da por hecho que dispone de un dominio de OpenSearch Service existente. Si no creó un dominio, siga los pasos de Creación y administración de dominios para crear uno.

Paso 1: crear la función Lambda

En este procedimiento, creará una función Lambda sencilla que servirá como destino para los mensajes de eventos de OpenSearch Service.

Para crear una función de Lambda de destino

1. Abra la consola de AWS Lambda en https://console.aws.amazon.com/lambda/.
2. Seleccione Crear función y Crear desde cero.
3. En Nombre de la función, ingrese event-handler.

4. En Tiempo de ejecución, seleccione Python 3.8.
5. Seleccione Crear función.
6. En la sección Código de función, edite el código de muestra de tal modo que coincida con el siguiente ejemplo:

```
import json

def lambda_handler(event, context):
    if event["source"] != "aws.es":
        raise ValueError("Function only supports input from events with a source
 type of: aws.es")

    print(json.dumps(event))
```

   Se trata de una función simple de Python 3.8 que imprime los eventos enviados por OpenSearch Service. Si todo está configurado correctamente, al final de este tutorial verá los detalles de los eventos del flujo de registros de Registros de CloudWatch asociado a esta función de Lambda.
7. Seleccione Implementar.

Paso 2: registrar una regla de eventos

En este paso, creará una regla de EventBridge que captura eventos de los dominios de OpenSearch Service. Esta regla captura todos los eventos dentro de la cuenta, en la cual se define. Los propios mensajes de eventos contienen información acerca de la fuente del evento, incluido el dominio desde el que se originó. Puede utilizar esta información para filtrar y ordenar eventos de manera programática.

Para crear una regla de EventBridge

1. Abra la consola de EventBridge en https://console.aws.amazon.com/events/.
2. Seleccione Crear regla.
3. Nombre la regla event-rule.
4. Seleccione Siguiente.
5. En el patrón de eventos, seleccione Servicios de AWS, Amazon OpenSearch Service y Todos los eventos. Este patrón se aplica a todos los dominios de OpenSearch Service y a todos los eventos de OpenSearch Service. También puede crear un patrón más específico para filtrar algunos resultados.

6. Pulse Siguiente.
7. Para el destino, seleccione Función de Lambda. En el menú desplegable de funciones, elija event-handler.
8. Pulse Siguiente.
9. Omita las etiquetas y vuelva a pulsar Siguiente.
10. Revise la configuración y elija Crear regla.

Paso 3: probar la configuración

La próxima vez que reciba una notificación en la sección Notificaciones de la consola de OpenSearch Service, si todo está configurado correctamente, la función de Lambda se activa y escribe los datos del evento en un flujo de registros de Registros de CloudWatch para la función.

Para probar la configuración

1. Abra la consola de CloudWatch en https://console.aws.amazon.com/cloudwatch/.
2. En el panel de navegación, seleccione Registros y, a continuación, seleccione el grupo de registro de la función de Lambda (por ejemplo, /aws/lambda/event-handler).
3. Seleccione una secuencia de registro para ver los datos de los eventos.

Tutorial: Sending Amazon SNS alerts for available software updates

En este tutorial, puede configurar una regla de eventos de Amazon EventBridge que captura notificaciones de actualizaciones disponibles del software de servicio en Amazon OpenSearch Service y envía una notificación por email a través de Amazon Simple Notification Service (Amazon SNS).

Requisitos previos

En este tutorial se da por hecho que dispone de un dominio de OpenSearch Service existente. Si no creó un dominio, siga los pasos de Creación y administración de dominios para crear uno.

Paso 1: crear y suscribirse a un tema de Amazon SNS

Configure un tema de Amazon SNS para servir como destino de eventos para la nueva regla de eventos.

Para crear un destino de Amazon SNS

1. Abra la consola de Amazon SNS en https://console.aws.amazon.com/sns/v3/home.
2. Seleccione Temas y Crear tema.
3. Para el tipo de tarea, seleccione Estándar y nombre el trabajo software-update.
4. Seleccione Crear nuevo tema.
5. Una vez creado el tema, seleccione Crear suscripción.
6. En Protocolo, seleccione Correo electrónico. En Punto de conexión, ingrese una dirección de email a la que actualmente tenga acceso y elija Crear suscripción.
7. Consulte su cuenta de email y espere para recibir un mensaje de correo electrónico de confirmación de la suscripción. Cuando lo reciba, seleccione Confirmar suscripción.

Paso 2: registrar una regla de eventos

A continuación, registre una regla de eventos que solo capture los eventos de actualización del software de servicio.

Para crear una regla de eventos

1. Abra la consola de EventBridge en https://console.aws.amazon.com/events/.
2. Seleccione Crear regla.
3. Nombre la regla softwareupdate-rule.
4. Elija Siguiente.
5. Para el patrón de evento, seleccione servicios de AWS, Amazon OpenSearch Service (cluster administrado) y Amazon OpenSearch Service Software Update Notification. Este patrón coincide con cualquier evento de actualización de software de OpenSearch Service. Para obtener más información sobre patrones de eventos, consulte Patrones de eventos de Amazon EventBridge en la Guía del usuario de Amazon EventBridge.
6. Si lo desea, puede filtrar solo por gravedades específicas. Para conocer las gravedades de cada evento, consulte the section called “Eventos de actualización del software de servicio”.
7. Seleccione Siguiente.
8. Para el destino, seleccione Tema de SNS y seleccione software-update.
9. Seleccione Siguiente.
10. Omita las etiquetas y elija Siguiente.
11. Revise la configuración de la regla y elija Crear regla.

La próxima vez que reciba una notificación de OpenSearch Service sobre una actualización del software de servicio disponible, si todo está configurado correctamente, Amazon SNS debería enviar una alerta por email sobre la actualización.

Monitoreo de las llamadas a la API de Amazon OpenSearch Service con AWS CloudTrail

Amazon OpenSearch Service se integra con AWS CloudTrail, un servicio que proporciona un registro de las medidas adoptadas por un usuario, un rol o un servicio AWS en OpenSearch Service. CloudTrail captura todas las llamadas de la API de configuración para OpenSearch Service como eventos.

> **Note**
>
> CloudTrail solo captura las llamadas a la API de configuración, como `CreateDomain` y `GetUpgradeStatus`. CloudTrail no captura llamadas a la API de OpenSearch, como `_search` y `_bulk`. Para estas llamadas, consulte the section called “Monitoreo de registros de auditoría”.

Entre las llamadas capturadas se incluyen las que se realizan desde la consola de OpenSearch Service, la AWS CLI o un SDK de AWS. Si crea un registro de seguimiento, puede habilitar la entrega continua de eventos de CloudTrail a un bucket de Amazon S3, incluidos los eventos para OpenSearch Service. Si no configura un registro de seguimiento, puede ver los eventos más recientes en la consola de CloudTrail en el Historial de eventos. Mediante la información recopilada por CloudTrail, puede determinar la solicitud que se realizó a OpenSearch Service, la dirección IP desde la cual se realizó, quién la realizó y cuándo, etc.

Para obtener más información sobre CloudTrail, consulte la Guía del usuario de AWS CloudTrail.

Información de Amazon OpenSearch Service en CloudTrail

CloudTrail se habilita en Cuenta de AWS cuando crea la cuenta. Cuando se produce una actividad en OpenSearch Service, esa actividad se registra en un evento de CloudTrail junto con otros eventos de servicio de AWS en el Event history (Historial de eventos). Puede ver, buscar y descargar los últimos eventos de la cuenta de Cuenta de AWS. Para obtener más información, consulte Ver eventos con el historial de eventos de CloudTrail.

Para mantener un registro continuo de los eventos de la cuenta de Cuenta de AWS, incluidos los eventos de OpenSearch Service, cree un registro de seguimiento. Un registro de seguimiento permite a CloudTrail enviar archivos de registro a un bucket de Amazon S3. De forma predeterminada, cuando se crea un registro de seguimiento en la consola, el registro de seguimiento se aplica a todas las Regiones de AWS. El registro de seguimiento registra los eventos de todas las regiones de la partición de AWS y envía los archivos de registro al bucket de Amazon S3 especificado. También es posible configurar otros servicios de AWS para analizar en profundidad y actuar en función de los datos de eventos recopilados en los registros de CloudTrail. Para más información, consulte los siguientes temas:

- Creación de una traza para su Cuenta de AWS
- AWS Integraciones de servicios de con registros de CloudTrail
- Configuración de notificaciones de Amazon SNS para CloudTrail
- Recibir archivos de registro de CloudTrail desde varias regiones y Recibir archivos de registro de CloudTrail desde varias cuentas

CloudTrail registra todas las acciones de la API de configuración de OpenSearch Service que se documentan en la referencia de API de Amazon OpenSearch Service.

Cada entrada de registro o evento contiene información sobre quién generó la solicitud. La información de identidad del usuario le ayuda a determinar lo siguiente:

- Si la solicitud se realizó con las credenciales raíz o del usuario de AWS Identity and Access Management (IAM).
- Si la solicitud se realizó con credenciales de seguridad temporales de un rol o fue un usuario federado
- si la solicitud la realizó otro servicio de AWS

Para obtener más información, consulte el Elemento userIdentity de CloudTrail.

Descripción de las entradas del archivo de registro de Amazon OpenSearch Service

Un registro de seguimiento es una configuración que habilita la entrega de eventos como archivos de registro en un bucket de Amazon S3 que especifique. Los archivos de registro de CloudTrail pueden contener una o varias entradas de registro. Un evento representa una solicitud específica realizada

desde un origen cualquiera y contiene información sobre la acción solicitada, la fecha y la hora de la acción, los parámetros de la solicitud, etc. Los archivos de registro de CloudTrail no rastrean el orden en la pila de las llamadas públicas a la API, por lo que estas no aparecen en ningún orden específico.

En el ejemplo que sigue se muestra una entrada de registro de CloudTrail que ilustra la operación `CreateDomain`:


```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
    "arn": "arn:aws:iam::123456789012:user/test-user",
    "accountId": "123456789012",
    "accessKeyId": "access-key",
    "userName": "test-user",
    "sessionContext": {
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2018-08-21T21:59:11Z"
      }
    },
    "invokedBy": "signin.amazonaws.com"
  },
  "eventTime": "2018-08-21T22:00:05Z",
  "eventSource": "es.amazonaws.com",
  "eventName": "CreateDomain",
  "awsRegion": "us-west-1",
  "sourceIPAddress": "123.123.123.123",
  "userAgent": "signin.amazonaws.com",
  "requestParameters": {
    "engineVersion": "OpenSearch_1.0",
    "clusterConfig": {
      "instanceType": "m4.large.search",
      "instanceCount": 1
    },
    "snapshotOptions": {
      "automatedSnapshotStartHour": 0
    },
    "domainName": "test-domain",
    "encryptionAtRestOptions": {},
    "eBSOptions": {
      "eBSEnabled": true,
```

```
      "volumeSize": 10,
      "volumeType": "gp2"
    },
    "accessPolicies": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":\"Allow
\",\"Principal\":{\"AWS\":[\"123456789012\"]},\"Action\":[\"es:*\"],\"Resource\":
\"arn:aws:es:us-west-1:123456789012:domain/test-domain/*\"}]}",
    "advancedOptions": {
      "rest.action.multi.allow_explicit_index": "true"
    }
  },
  "responseElements": {
    "domainStatus": {
      "created": true,
      "clusterConfig": {
        "zoneAwarenessEnabled": false,
        "instanceType": "m4.large.search",
        "dedicatedMasterEnabled": false,
        "instanceCount": 1
      },
      "cognitoOptions": {
        "enabled": false
      },
      "encryptionAtRestOptions": {
        "enabled": false
      },
      "advancedOptions": {
        "rest.action.multi.allow_explicit_index": "true"
      },
      "upgradeProcessing": false,
      "snapshotOptions": {
        "automatedSnapshotStartHour": 0
      },
      "eBSOptions": {
        "eBSEnabled": true,
        "volumeSize": 10,
        "volumeType": "gp2"
      },
      "engineVersion": "OpenSearch_1.0",
      "processing": true,
      "aRN": "arn:aws:es:us-west-1:123456789012:domain/test-domain",
      "domainId": "123456789012/test-domain",
      "deleted": false,
      "domainName": "test-domain",
```

```
      "accessPolicies": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":\"Allow
\",\"Principal\":{\"AWS\":\"arn:aws:iam::123456789012:root\"},\"Action\":\"es:*\",
\"Resource\":\"arn:aws:es:us-west-1:123456789012:domain/test-domain/*\"}]}"
    }
  },
  "requestID": "12345678-1234-1234-1234-987654321098",
  "eventID": "87654321-4321-4321-4321-987654321098",
  "eventType": "AwsApiCall",
  "recipientAccountId": "123456789012"
}
```

Seguridad en Amazon OpenSearch Service

La seguridad en la nube de AWS es la máxima prioridad. Como cliente de AWS, se beneficia de una arquitectura de red y un centro de datos que se han diseñado para satisfacer los requisitos de seguridad de las organizaciones más exigentes.

La seguridad es una responsabilidad compartida entre AWS y el usuario. El modelo de responsabilidad compartida la describe como seguridad de la nube y seguridad en la nube:

- Seguridad de la nube: AWS es responsable de proteger la infraestructura que ejecuta los servicios de AWS en la nube de AWS. AWS también proporciona servicios que puede utilizar de forma segura. Auditores externos prueban y verifican periódicamente la eficacia de nuestra seguridad en el marco de los programas de conformidad de AWS. Para obtener más información acerca de los programas de conformidad que se aplican a Amazon OpenSearch Service, consulte Servicios de AWS en el ámbito del programa de conformidad.
- Seguridad en la nube: su responsabilidad se determina según el servicio de AWSque utiliza. También eres responsable de otros factores, incluida la confidencialidad de los datos, los requisitos de la empresa y la legislación y la normativa aplicables.

Esta documentación lo ayuda a comprender cómo aplicar el modelo de responsabilidad compartida cuando utiliza OpenSearch Service. En los siguientes temas, se muestra cómo configurar OpenSearch Service para cumplir sus objetivos de seguridad y conformidad. También puede aprender a utilizar otros servicios de AWS que lo ayudan a monitorear y proteger los recursos de OpenSearch Service.

Temas

- Protección de los datos en Amazon OpenSearch Service
- Identity and Access Management en Amazon OpenSearch Service
- Prevención de la sustitución confusa entre servicios
- Control de acceso detallado en Amazon OpenSearch Service
- Validación de conformidad para Amazon OpenSearch Service
- Resiliencia en Amazon OpenSearch Service
- Autenticación y autorización de JWT para Amazon OpenSearch Service
- Seguridad de la infraestructura en Amazon OpenSearch Service
- Autenticación SAML para OpenSearch Dashboards

- Compatibilidad de la propagación de identidades de confianza del Centro de identidades de IAM con OpenSearch
- Configuración de la autenticación de Amazon Cognito para OpenSearch Dashboards
- Uso de roles vinculados a servicios para Amazon OpenSearch Service.

Protección de los datos en Amazon OpenSearch Service

El modelo de responsabilidad compartida de AWS se aplica a la protección de los datos en Amazon OpenSearch Service. Como se describe en este modelo, AWS es responsable de proteger la infraestructura global que ejecuta toda la Nube de AWS. Eres responsable de mantener el control sobre el contenido alojado en esta infraestructura. También eres responsable de las tareas de administración y configuración de seguridad para los Servicios de AWS que utiliza. Para obtener más información sobre la privacidad de los datos, consulta las Preguntas frecuentes sobre la privacidad de datos. Para obtener información sobre la protección de datos en Europa, consulta la publicación de blog sobre el Modelo de responsabilidad compartida de AWS y GDPR en el Blog de seguridad de AWS.

Con fines de protección de datos, recomendamos proteger las credenciales de la Cuenta de AWS y configurar cuentas de usuario individuales con AWS IAM Identity Center o AWS Identity and Access Management (IAM). De esta manera, solo se otorgan a cada usuario los permisos necesarios para cumplir sus obligaciones laborales. También recomendamos proteger sus datos de la siguiente manera:

- Utiliza la autenticación multifactor (MFA) en cada cuenta.
- Utiliza SSL/TLS para comunicarse con los recursos de AWS. Se recomienda el uso de TLS 1.2 y recomendamos TLS 1.3.
- Configure los registros de API y de actividad de los usuarios con AWS CloudTrail. Para obtener información sobre cómo utilizar registros de seguimiento de CloudTrail para capturar actividades de AWS, consulta Working with CloudTrail trails en la Guía del usuario de AWS CloudTrail.
- Utiliza las soluciones de cifrado de AWS, junto con todos los controles de seguridad predeterminados dentro de los servicios de Servicios de AWS.
- Utiliza servicios de seguridad administrados avanzados, como Amazon Macie, que lo ayuden a detectar y proteger los datos confidenciales almacenados en Amazon S3.
- Si necesita módulos criptográficos validados FIPS 140-3 al acceder a AWS a través de una interfaz de línea de comandos o una API, utiliza un punto de conexión de FIPS. Para obtener

más información sobre los puntos de conexión de FIPS disponibles, consulta Estándar de procesamiento de la información federal (FIPS) 140-3.

Se recomienda encarecidamente no introducir nunca información confidencial o sensible, como por ejemplo, direcciones de correo electrónico de clientes, en etiquetas o campos de formato libre, tales como el campo Nombre. Tampoco debe hacerlo cuando trabaje con OpenSearch Service u otros Servicios de AWS mediante la consola, la API, la AWS CLI o los AWS SDK. Cualquier dato que ingrese en etiquetas o campos de formato libre utilizados para nombres se pueden emplear para los registros de facturación o diagnóstico. Si proporciona una URL a un servidor externo, recomendamos encarecidamente que no incluya información de credenciales en la URL a fin de validar la solicitud para ese servidor.

Cifrado de datos en reposo para Amazon OpenSearch Service

Los dominios de OpenSearch Service ofrecen cifrado de datos en reposo, una característica de seguridad que ayuda a prevenir el acceso no autorizado a los datos. La característica utiliza AWS Key Management Service (AWS KMS) para almacenar y administrar sus claves de cifrado y el algoritmo estándar de cifrado avanzado con claves de 256 bits (AES-256) para realizar el cifrado. Si está habilitada, la característica cifra los siguientes aspectos de un dominio:

- Todos los índices (incluidos los de almacenamiento UltraWarm)
- Registros de OpenSearch
- Archivos de intercambio
- Todos los demás datos del directorio de la aplicación
- Instantáneas automatizadas

Los siguientes elementos no se cifran cuando habilita el cifrado de datos en reposo, pero puede realizar pasos adicionales para protegerlos:

- Instantáneas manuales: actualmente, no puede utilizar claves de AWS KMS para cifrar instantáneas manuales. No obstante, puede utilizar el cifrado del lado del servidor con claves administradas por S3 o claves de KMS para cifrar el bucket que utiliza como repositorio de instantáneas. Para obtener instrucciones, consulte the section called “Registrar un repositorio de instantáneas manuales”.
- Registros lentos y registros de errores: si publica registros y desea cifrarlos, puede cifrar su grupo de registros de CloudWatch Logs con la misma clave de AWS KMS que utiliza para el dominio de

OpenSearch Service. Para obtener más información, consulte Cifrar datos de registro en Registros de CloudWatch mediante AWS Key Management Service en la Guía del usuario de Registros de Amazon CloudWatch.

> **Note**
>
> No puede habilitar el cifrado en reposo en un dominio existente si UltraWarm o el almacenamiento en frío están habilitados en el dominio. Primero debe deshabilitar UltraWarm o el almacenamiento en frío, habilitar el cifrado en reposo y, a continuación, volver a habilitar UltraWarm o el almacenamiento en frío. Si desea retener los índices en UltraWarm o en almacenamiento en frío, debe moverlos a un almacenamiento en caliente antes de deshabilitar UltraWarm o almacenamiento en frío.

OpenSearch Service solo admite claves de cifrado de KMS simétricas, no asimétricas. Para conocer cómo crear claves simétricas, consulte Crear una clave de KMS en la Guía para desarrolladores de AWS Key Management Service.

Independientemente de si el cifrado en reposo está habilitado, todos los dominios cifran automáticamente paquetes personalizados con claves administradas por AES-256 y OpenSearch Service.

Permisos

Para utilizar la consola de OpenSearch Service a fin de configurar el cifrado de datos en reposo, debe contar con permisos de lectura para AWS KMS, como la siguiente política con base en identidades:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "kms:List*",
        "kms:Describe*"
      ],
```

```
      "Resource": "*"
    }
  ]
}
```

Si desea utilizar una clave que no sea la de AWS, también debe disponer de permisos a fin de crear concesiones para la clave. Estos permisos normalmente suelen adoptar la forma de una política con base en recursos que especifica al crear la clave.

Si desea conservar la clave exclusiva para OpenSearch Service, puede agregar la condición kms:ViaService a esa política de claves:

```
"Condition": {
  "StringEquals": {
    "kms:ViaService": "es.us-west-1.amazonaws.com"
  },
  "Bool": {
    "kms:GrantIsForAWSResource": "true"
  }
}
```

Para obtener más información, consulte Políticas de claves en AWS KMS en la Guía para desarrolladores de AWS Key Management Service.

Habilitar el cifrado de datos en reposo

El cifrado de datos en reposo en dominios nuevos requiere la versión 5.1 de OpenSearch o Elasticsearch o versiones posteriores. Para habilitarlo en dominios existentes, se requiere la versión 6.7 de OpenSearch o Elasticsearch o versiones posteriores.

Para habilitar el cifrado de los datos en reposo (consola)

1. Abra el dominio en la consola de AWS, a continuación, elija Actions (Acciones) y Edit security configuration (Editar la configuración de seguridad).
2. En Cifrado, seleccione Habilitar el cifrado de datos en reposo.
3. Seleccione una clave AWS KMS para usar y, a continuación, seleccione Guardar cambios.

También puede habilitar el cifrado a través de la API de configuración. La siguiente solicitud permite el cifrado de datos en reposo en un dominio existente:

```
{
   "ClusterConfig":{
      "EncryptionAtRestOptions":{
         "Enabled": true,
         "KmsKeyId":"arn:aws:kms:us-east-1:123456789012:alias/my-key"
      }
   }
}
```

Clave KMS desactivada o eliminada

Si desactiva o elimina la clave que utilizó para cifrar un dominio, el dominio deja de estar accesible. OpenSearch Service le envía una notificación informándole de que no puede acceder a la clave KMS. Vuelva a habilitar la clave inmediatamente para acceder a su dominio.

El equipo del Servicio de OpenSearch no puede ayudarte a recuperar tus datos si tu clave se ha borrado. AWS KMS borra las claves sólo después de un periodo de espera de al menos siete días. Si su clave está pendiente de eliminación, cancele la eliminación o tome una instantánea manual del dominio para evitar la pérdida de datos.

Deshabilitar el cifrado de datos en reposo

Después de configurar un dominio para cifrar los datos en reposo, no puede desactivar la configuración. En su lugar, puede realizar una instantánea manual del dominio existente, crear otro dominio, migrar los datos y eliminar el dominio antiguo.

Monitorear dominios que cifran los datos en reposo

Los dominios que cifran los datos en reposo tienen dos métricas adicionales: `KMSKeyError` y `KMSKeyInaccessible`. Estas métricas solo aparecen si el dominio se encuentra con un problema con la clave de cifrado. Para obtener una descripción completa de estas métricas, consulte the section called “Métricas de clúster”. Puede verlas con la consola de OpenSearch Service o la consola de Amazon CloudWatch.

> **Tip**
>
> Cada métrica representa un problema importante para un dominio, por lo que recomendamos que cree alarmas de CloudWatch para ambas. Para obtener más información, consulte the section called “Alarmas de CloudWatch recomendadas”.

Otras consideraciones

- La rotación automática de claves conserva las propiedades de sus claves de AWS KMS, por lo que la rotación no tiene efecto sobre su capacidad de acceder a los datos de OpenSearch. Los dominios cifrados de OpenSearch Service no admiten la rotación manual de claves, lo que implica la creación de una nueva clave y la actualización de cualquier referencia a la antigua clave. Para obtener más información, consulte Rotar claves de AWS KMS en la Guía para desarrolladores de AWS Key Management Service.
- Algunos tipos de instancias no admiten el cifrado de datos en reposo. Para obtener más información, consulte the section called “Tipos de instancias admitidas”.
- Los dominios que cifran los datos en reposo utilizan otro nombre de repositorio para sus instantáneas automatizadas. Para obtener más información, consulte the section called “Restaurar instantáneas”.
- Si bien recomendamos encarecidamente habilitar el cifrado en reposo, puede agregar sobrecarga de CPU adicional y unos pocos milisegundos de latencia. Sin embargo, la mayoría de los casos de uso no son sensibles a estas diferencias y la magnitud del impacto depende de la configuración del clúster, los clientes y el perfil de uso.

Cifrado de nodo a nodo para Amazon OpenSearch Service

El cifrado de nodo a nodo proporciona una capa de seguridad adicional además de las características predeterminadas de Amazon OpenSearch Service.

Cada dominio de OpenSearch Service, independientemente de si el dominio utiliza el acceso a la VPC, reside en su propia VPC dedicada. Esta arquitectura impide que los atacantes potenciales intercepten el tráfico entre los nodos de OpenSearch y mantiene el clúster seguro. Sin embargo, de forma predeterminada, el tráfico de la VPC está sin cifrar. El cifrado de nodo a nodo habilita el cifrado TLS 1.2 para todas las comunicaciones dentro de la VPC.

Si envía datos a OpenSearch Service a través de HTTPS, el cifrado de nodo a nodo ayuda a garantizar que sus datos permanezcan cifrados a medida que OpenSearch los distribuye (y redistribuye) en todo el clúster. Si los datos llegan sin cifrar a través de HTTP, OpenSearch Service los cifra después de que alcancen el clúster. Puede exigir que todo el tráfico al dominio llegue a través de HTTPS mediante la consola, la AWS CLI o la API de configuración.

El cifrado de nodo a nodo es obligatorio para la habilitación del control de acceso detallado.

Habilitar el cifrado de nodo a nodo

El cifrado de nodo a nodo en dominios nuevos requiere cualquier versión de OpenSearch, Elasticsearch 6.0 o posterior. Habilitar el cifrado de nodo a nodo en dominios existentes requiere cualquier versión de OpenSearch, Elasticsearch 6.7 o posterior. Seleccione el dominio existente en la consola de AWS, Acciones y Editar la configuración de seguridad.

También puede utilizar la AWS CLI o la API de configuración. Para obtener más información, consulte la Referencia de los comandos de la AWS CLI y la Referencia de la API de OpenSearch Service.

Deshabilitar el cifrado de nodo a nodo

Después de configurar un dominio para utilizar el cifrado de nodo a nodo, no puede desactivar la configuración. En su lugar, puede realizar una instantánea manual del dominio cifrado, crear otro dominio, migrar los datos y eliminar el dominio antiguo.

Identity and Access Management en Amazon OpenSearch Service

Amazon OpenSearch Service ofrece varias maneras de controlar el acceso a sus dominios. En este tema, se tratan los diversos tipos de política, la forma en que interactúan entre sí y cómo crear sus propias políticas personalizadas.

> **Important**
>
> La compatibilidad con la VPC presenta algunas consideraciones adicionales al control del acceso de OpenSearch Service. Para más información, consulte the section called “Acerca de las políticas de acceso en los dominios de VPC”.

Tipos de políticas

OpenSearch Service admite tres tipos de políticas de acceso:

- the section called “Políticas basadas en recursos”
- the section called “Políticas basadas en identidad”
- the section called “Políticas basadas en IP”

Políticas basadas en recursos

Al crear un dominio, agrega una política basada en recursos, a veces denominada política de acceso al dominio. Estas políticas especifican qué acciones puede realizar una entidad principal en los subrecursos del dominio (con la excepción de la búsqueda entre clústeres). Entre los subrecursos se incluyen los índices y las API de OpenSearch. El elemento de la política JSON Principal (Entidad principal) en IAM especifica las cuentas, los usuarios o los roles que tienen permitido el acceso. El elemento de la política JSON Resource (Recurso) especifica los subrecursos a los que pueden obtener acceso estas entidades principales.

Por ejemplo, la siguiente política basada en recursos concede a `test-user` acceso completo (`es:*`) a los subrecursos en `test-domain`:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "arn:aws:iam::123456789012:user/test-user"
        ]
      },
      "Action": [
        "es:*"
      ],
      "Resource": "arn:aws:es:us-west-1:987654321098:domain/test-domain/*"
    }
  ]
}
```

Dos consideraciones importantes se aplican a esta política:

- Estos privilegios se aplican únicamente a este dominio. A menos que cree políticas similares en otros dominios, `test-user` solo puede acceder a `test-domain`.

- El `/*` final en el elemento `Resource` es significativo e indica que las políticas basadas en recursos solo se aplican a los subrecursos del dominio, no al propio dominio. En las políticas basadas en recursos, la acción `es:*` es equivalente a `es:ESHttp*`.

  Por ejemplo, `test-user` puede realizar solicitudes frente a un índice (`GET https://search-test-domain.us-west-1.es.amazonaws.com/test-index`), pero no puede actualizar la configuración del dominio (`POST https://es.us-west-1.amazonaws.com/2021-01-01/opensearch/domain/test-domain/config`). Observe la diferencia entre los dos puntos de conexión. El acceso a la API de configuración requiere una política basada en identidad.

Puede especificar un nombre de índice parcial agregando un comodín. Este ejemplo identifica cualquier índice que empiecen con `commerce`:

```
arn:aws:es:us-west-1:987654321098:domain/test-domain/commerce*
```

En este caso, el comodín significa que `test-user` puede realizar solicitudes a índices dentro de `test-domain` que tengan nombres que comiencen con `commerce`.

Para restringir aún más `test-user`, puede aplicar la siguiente política:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "arn:aws:iam::123456789012:user/test-user"
        ]
      },
      "Action": [
        "es:ESHttpGet"
      ],
      "Resource": "arn:aws:es:us-west-1:987654321098:domain/test-domain/commerce-data/_search"
    }
  ]
```

```
}
```

Ahora `test-user` puede realizar solo una operación: búsquedas del índice `commerce-data`. Todos los demás índices del dominio son inaccesibles y, sin permisos para utilizar las acciones `es:ESHttpPut` o `es:ESHttpPost`, `test-user` no puede agregar ni modificar documentos.

A continuación, podría decidir configurar un rol para usuarios avanzados. Esta política concede a `power-user-role` acceso a los métodos HTTP GET y PUT para todos los URI del índice:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "arn:aws:iam::123456789012:role/power-user-role"
        ]
      },
      "Action": [
        "es:ESHttpGet",
        "es:ESHttpPut"
      ],
      "Resource": "arn:aws:es:us-west-1:987654321098:domain/test-domain/commerce-
data/*"
    }
  ]
}
```

Si el dominio se encuentra en una VPC o utiliza un control de acceso detallado, puede utilizar una política de acceso al dominio abierto. De lo contrario, la política de acceso al dominio debe contener alguna restricción, ya sea por entidad principal o dirección IP.

Para obtener información sobre todas las acciones disponibles, consulte the section called “Referencia de los elementos de las políticas”. Para obtener un control mucho más pormenorizado sobre sus datos, utilice una política de acceso al dominio abierto con control de acceso detallado.

Políticas basadas en identidad

A diferencia de las políticas basadas en recursos, que forman parte de cada dominio en OpenSearch Service, la políticas basadas en identidad se adjuntan a usuarios o roles mediante el servicio de AWS Identity and Access Management (IAM). Al igual que las políticas basadas en recursos, las políticas basadas en identidad especifican quién puede obtener acceso a un servicio, las acciones que puede realizar y, si corresponde, los recursos en los que se pueden realizar dichas acciones.

Aunque en realidad no tienen por qué serlo, las políticas basadas en identidad suelen ser más genéricas. Normalmente, solo determinan las acciones de la API de configuración que un usuario puede realizar. Después de poner en vigor estas políticas, puede utilizar políticas basadas en recursos (o control de acceso detallado) en OpenSearch Service para ofrecer a los usuarios acceso a los índices y las API de OpenSearch.

> **Note**
>
> Los usuarios con la política `AmazonOpenSearchServiceReadOnlyAccess` administrada por AWS no pueden ver el estado del clúster en la consola. Para permitir que vean el estado del clúster (y otros datos de OpenSearch), agregue la acción `es:ESHttpGet` a una política de acceso y adjúntela a sus cuentas o roles.

Dado que las políticas basadas en identidad se adjuntan a usuarios o roles (entidades principales), el JSON no especifica una entidad principal. La siguiente política concede acceso a las acciones que comienzan por `Describe` y `List`. Esta combinación de acciones proporciona acceso de solo lectura a las configuraciones de dominio, pero no a los datos almacenados en el dominio:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Action": [
        "es:Describe*",
        "es:List*"
      ],
      "Effect": "Allow",
      "Resource": "*"
```

```
    }
  ]
}
```

Un administrador puede tener acceso completo a OpenSearch Service y todos los datos almacenados en todos los dominios:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Action": [
        "es:*"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

Las políticas basadas en identidad permiten utilizar etiquetas para controlar el acceso a la API de configuración. La siguiente política, por ejemplo, permite a las entidades principales adjuntas ver y actualizar la configuración de un dominio si el dominio tiene la etiqueta `team:devops`:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [{
    "Action": [
      "es:UpdateDomainConfig",
      "es:DescribeDomain",
      "es:DescribeDomainConfig"
    ],
    "Effect": "Allow",
    "Resource": "*",
    "Condition": {
```

```
      "ForAnyValue:StringEquals": {
        "aws:ResourceTag/team": [
          "devops"
        ]
      }
    }
  }]
}
```

Ahora puede utilizar etiquetas para controlar el acceso a la API de OpenSearch. Las políticas basadas en etiquetas para la API de OpenSearch solo se aplican a los métodos HTTP. Por ejemplo, la siguiente política permite a las entidades principales adjuntas enviar solicitudes GET y PUT a la API de OpenSearch si el dominio tiene la etiqueta `environment:production`:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [{
    "Action": [
      "es:ESHttpGet",
      "es:ESHttpPut"
    ],
    "Effect": "Allow",
    "Resource": "*",
    "Condition": {
      "ForAnyValue:StringEquals": {
        "aws:ResourceTag/environment": [
          "production"
        ]
      }
    }
  }]
}
```

Para tener un control más detallado de la API de OpenSearch, considere utilizar el control de acceso detallado.

> **Note**
>
> Después de agregar una o más API de OpenSearch a cualquier política basada en etiquetas, debe realizar una sola operación de etiquetas (como agregar, eliminar o modificar una etiqueta) para que los cambios surtan efecto en un dominio. Debe estar en el software de servicio R20211203 o posterior para incluir operaciones de API de OpenSearch en políticas basadas en etiquetas.

OpenSearch Service admite las claves de condición globales `RequestTag` y `TagKeys` para la API de configuración, pero no para la API de OpenSearch. Estas condiciones solo se aplican a las llamadas a la API que incluyen etiquetas dentro de la solicitud, como `CreateDomain`, `AddTags` y `RemoveTags`. La siguiente política permite a las entidades principales adjuntas crear dominios, pero solo si incluyen la etiqueta `team:it` en la solicitud:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Action": [
      "es:CreateDomain",
      "es:AddTags"
    ],
    "Resource": "*",
    "Condition": {
      "StringEquals": {
        "aws:RequestTag/team": [
          "it"
        ]
      }
    }
  }
}
```

Para obtener más información sobre el uso de etiquetas para el control de acceso y las diferencias entre las políticas basadas en recursos y las políticas basadas en identidad, consulte Definición de permisos en función de los atributos con la autorización de ABAC en la Guía del usuario de IAM.

Políticas basadas en IP

Las políticas basadas en IP restringen el acceso a un dominio a una o más direcciones IP o bloques de CIDR. Técnicamente, las políticas basadas en IP no son un tipo de política distinto. En lugar de ello, son solo políticas basadas en recursos que especifican una entidad principal anónima e incluyen una Condición especial. Para obtener información, consulte Elementos de la política JSON de IAM: Condición en la Guía del usuario de IAM.

El atractivo principal de las políticas basadas en IP es que permiten solicitudes sin firmar en un dominio de OpenSearch Service, lo que permite utilizar clientes como curl y OpenSearch Dashboards o acceder al dominio a través de un servidor proxy. Para más información, consulte the section called “Utilización de un proxy para acceder a OpenSearch Service desde Dashboards”.

> **Note**
>
> Si ha habilitado acceso VPC para su dominio, no puede configurar una política basada en IP. En su lugar, puede utilizar `security groups` para controlar qué direcciones IP pueden tener acceso al dominio. Para obtener más información, consulte los temas siguientes:
>
> - the section called “Acerca de las políticas de acceso en los dominios de VPC”
> - Controlar el tráfico hacia los recursos de AWS mediante grupos de seguridad en la Guía del usuario de Amazon VPC

La siguiente política concede a todas las solicitudes HTTP que se originan en el intervalo de IP especificado acceso a `test-domain`:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
      },
      "Action": [
        "es:ESHttp*"
```

```
      ],
      "Condition": {
        "IpAddress": {
          "aws:SourceIp": [
            "192.0.2.0/24"
          ]
        }
      },
      "Resource": "arn:aws:es:us-west-1:987654321098:domain/test-domain/*"
    }
  ]
}
```

Si su dominio tiene un punto de conexión público y no utiliza un control de acceso detallado, recomendamos combinar direcciones IP y entidades principales de IAM. Esta política concede acceso HTTP `test-user` únicamente si la solicitud se origina en el intervalo IP especificado:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Principal": {
      "AWS": [
        "arn:aws:iam::987654321098:user/test-user"
      ]
    },
    "Action": [
      "es:ESHttp*"
    ],
    "Condition": {
      "IpAddress": {
        "aws:SourceIp": [
          "192.0.2.0/24"
        ]
      }
    },
    "Resource": "arn:aws:es:us-west-1:987654321098:domain/test-domain/*"
  }]
}
```

Realizar y firmar solicitudes de OpenSearch Service

Incluso si configura una política de acceso basada en recursos completamente abierta, todas las solicitudes a la API de configuración de OpenSearch Service deben estar firmadas. Si las políticas especifican usuarios o roles de IAM, las solicitudes a las API de OpenSearch también deben firmarse mediante Signature Version 4 de AWS. El método de firma difiere en función de la API:

- Para realizar llamadas a la API de configuración de OpenSearch Service, recomendamos que utilice uno de los SDK de AWS. Los SDK simplifican en gran medida el proceso y pueden ahorrar mucho tiempo en comparación con la creación y firma de sus propias solicitudes. Los puntos de conexión de la API de configuración utilizan el siguiente formato:

  ```
  es.region.amazonaws.com/2021-01-01/
  ```

  Por ejemplo, la siguiente solicitud introduce un cambio de configuración en el dominio `movies`, pero es preciso identificarse (no recomendado):

  ```
  POST https://es.us-east-1.amazonaws.com/2021-01-01/opensearch/domain/movies/config
  {
    "ClusterConfig": {
      "InstanceType": "c5.xlarge.search"
    }
  }
  ```

  Si utiliza uno de los SDK como Boto 3, el SDK se encarga automáticamente de la firma de solicitudes:

  ```
  import boto3

  client = boto3.client(es)
  response = client.update_domain_config(
    DomainName='movies',
    ClusterConfig={
      'InstanceType': 'c5.xlarge.search'
    }
  )
  ```

  Para ver un código de muestra de Java, consulte the section called “Utilizar los SDK de AWS”.

- Para realizar llamadas a las API de OpenSearch, debe firmar sus propias solicitudes. Las API de OpenSearch utilizan el formato siguiente:

  ```
  domain-id.region.es.amazonaws.com
  ```

  Por ejemplo, la siguiente solicitud busca en el índice `movies` de thor:

  ```
  GET https://my-domain.us-east-1.es.amazonaws.com/movies/_search?q=thor
  ```

> **Note**
>
> El servicio ignora los parámetros pasados en las direcciones URL de las solicitudes HTTP POST firmadas con Signature Version 4.

Cuando las políticas chocan

Surgen complejidades cuando las políticas están en desacuerdo o no hacen mención explícita a un usuario. Cómo funciona IAM en la Guía del usuario de IAM ofrece un breve resumen de la lógica de evaluación de políticas:

- De forma predeterminada, se deniegan todas las solicitudes.
- Un permiso explícito anula esta opción predeterminada.
- Una denegación explícita anula cualquier permiso concedido.

Por ejemplo, si una política basada en recursos concede acceso a un subrecurso del dominio (una API o índice de OpenSearch), pero una política basada en identidad deniega el acceso, se denegará el acceso. Si una política basada en identidad concede acceso y una política basada en recursos no especifica si debe tener acceso o no, tiene permitido el acceso. Consulte la siguiente tabla de políticas que se entrecruzan para acceder a un resumen completo de resultados de subrecursos del dominio.

	Permitido en política basada en recursos	Denegado en política basada en recursos	Ni permitido ni denegado en política basada en recursos
Allowed in identity-based policy	Permitir	Deny	Allow
Denied in identity-based policy	Deny	Deny	Deny
Neither allowed nor denied in identity-based policy	Allow	Deny	Deny

Referencia de los elementos de las políticas

OpenSearch Service admite la mayoría de los elementos de las políticas de la Referencia de los elementos de políticas de IAM, a excepción de `NotPrincipal`. La siguiente tabla muestra los elementos más comunes.

Elemento de política JSON	Resumen
`Version`	La versión actual del idioma de la política es `2012-10-17` . Todas las políticas de acceso deben especificar este valor.
`Effect`	Este elemento especifica si la instrucción permite o deniega el acceso a las acciones especificadas. Los valores válidos son `Allow` o `Deny`.
`Principal`	Este elemento especifica la Cuenta de AWS o el usuario o rol de IAM que tiene permitido o denegado el acceso a un recurso, y puede adoptar varias formas: • Cuentas de AWS: `"Principal":{"AWS": ["123456789012"]}` o `"Principal":{"AWS": ["arn:aws:iam::123456789012:root"]}`

Elemento de política JSON	Resumen
	• Usuarios de IAM: `"Principal":{"AWS": ["arn:aws:iam::123456789012:user/test-user"]}` • Roles de IAM: `"Principal":{"AWS": ["arn:aws:iam::123456789012:role/test-role"]}` **Important** Especificar el comodín * habilita el acceso anónimo al dominio, lo que no se recomienda a menos que agregue una condición basada en IP, utilice la compatibilidad con la VPC o habilite un control de acceso detallado. Además, analice detenidamente las siguientes políticas para confirmar que no otorgan acceso amplio: • Políticas basadas en identidad asociadas a las entidades principales de AWS vinculadas (por ejemplo, los roles de IAM) • Políticas basadas en recursos adjuntas a recursos de AWS asociados (por ejemplo, claves de AWS Key Management Service [KMS])

Elemento de política JSON	Resumen
`Action`	OpenSearch Service utiliza acciones `ESHttp*` para métodos HTTP de OpenSearch. El resto de las acciones se aplican a la API de configura ción. Determinadas acciones `es:` admiten permisos de nivel de recursos. Por ejemplo, puede otorgar a un usuario permisos para eliminar un dominio particular sin dar a dicho usuario permisos para eliminar cualquier dominio. Otras acciones se aplican solo al propio servicio. Por ejemplo, `es:ListDomainNames` no tiene sentido en el contexto de un dominio único y, por lo tanto, requiere un comodín. Para obtener una lista de todas las acciones disponibles y si se aplican a los subrecursos del dominio (`test-domain/*`), a la configuración del dominio (`test-domain`) o solo al servicio (`*`), consulte Acciones, recursos y claves de condición para Amazon OpenSearch Service en la Referencia de autorización de servicios Las políticas basadas en recursos difieren de los permisos en el nivel de recursos. Las políticas basadas en recursos son políticas JSON completas que se adjuntan a dominios. Los permisos de nivel de recurso permiten restringir acciones a dominios o subrecursos particulares. En la práctica, puede pensar en permisos de nivel de recursos como parte opcional de una política basada en recursos o identidades. Aunque los permisos de nivel de recurso para `es:CreateDomain` puedan parecer poco intuitivos (después de todo, ¿por qué darle a un usuario permisos para crear un dominio que ya existe?), la utilización de un comodín permite aplicar un esquema de nomenclatura sencillo para sus dominios, por ejemplo `"Resource": "arn:aws:es:us-west-1:987654321098:domain/my-team-name-*"` . Por supuesto, nada impide que incluya acciones junto a elementos de recursos menos restrictivos, como las siguientes:

Elemento de política JSON	Resumen
	JSON `{ "Version":"2012-10-17", "Statement": [ { "Effect": "Allow", "Action": [ "es:ESHttpGet", "es:DescribeDomain" ], "Resource": "*" } ] }` Para obtener más información acerca de las acciones y recursos de emparejamiento, consulte el elemento `Resource` en esta tabla.

Elemento de política JSON	Resumen
`Condition`	OpenSearch Service admite la mayoría de las condiciones que se describen en Claves de contexto de condición globales de AWS en la Guía del usuario de IAM. Entre las excepciones a tener en cuenta, se incluye la clave `aws:PrincipalTag` , que OpenSearch Service no admite. Al configurar una política basada en IP, debe especificar las direcciones IP o bloque de CIDR como condición, como en el ejemplo siguiente: `"Condition": { "IpAddress": { "aws:SourceIp": [ "192.0.2.0/32" ] } }` Como se señala en the section called “Políticas basadas en identidad ”, las claves de condición `aws:ResourceTag` , `aws:RequestTag` y `aws:TagKeys` solo se aplican a la API de configuración, además de a las API de OpenSearch.

Elemento de política JSON	Resumen

`Resource`	OpenSearch Service utiliza elementos `Resource` en tres formas básicas:
	
	• Para las acciones que se aplican al propio OpenSearch Service, como `es:ListDomainNames` , o para permitir el acceso completo, utilice la siguiente sintaxis:
	
	`"Resource": "*"`
	
	• Para las acciones que implican la configuración de un dominio, como `es:DescribeDomain` , puede utilizar la siguiente sintaxis:
	
	`"Resource": "arn:aws:es:` *region*`:`*aws-account-id*`:domain/`*domain-name*`"`
	
	• Para las acciones que se aplican a los subrecursos de un dominio, como `es:ESHttpGet` , puede utilizar la siguiente sintaxis:
	
	`"Resource": "arn:aws:es:` *region*`:`*aws-account-id*`:domain/`*domain-name*`/*"`
	
	No es necesario que utilice un comodín. OpenSearch Service permite definir una política de acceso diferente para cada API o índice de OpenSearch. Por ejemplo, podría limitar los permisos de un usuario al índice `test-index` :
	
	`"Resource": "arn:aws:es:` *region*`:`*aws-account-id*`:domain/`*domain-name*`/test-index"`
	
	En lugar de acceso completo a `test-index` , quizá prefiera limitar la política a solo la API de búsqueda:
	
	`"Resource": "arn:aws:es:` *region*`:`*aws-account-id*`:domain/`*domain-name*`/test-index/_search"`

Elemento de política JSON	Resumen
	Incluso puede controlar el acceso a documentos individuales: `"Resource": "arn:aws:es:  region:aws-account-id:domain/domain-name /test-index/test-type/1"` En esencia, si OpenSearch expresa el subrecurso como un URI, puede controlar el acceso a él mediante una política de acceso. Para obtener un mayor control sobre los recursos a los que puede obtener acceso un usuario, consulte the section called “Control de acceso detallado”. Para conocer detalles sobre qué acciones admiten los permisos en el nivel de recursos, consulte el elemento `Action` en esta tabla.

Opciones avanzadas y consideraciones de la API

OpenSearch Service tiene varias opciones avanzadas, una de las cuales tiene implicaciones sobre el control de acceso: `rest.action.multi.allow_explicit_index`. En su configuración predeterminada como verdadero, permite a los usuarios omitir los permisos de subrecursos en determinadas circunstancias.

Por ejemplo, tomemos la siguiente política basada en recursos:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "arn:aws:iam::123456789012:user/test-user"
        ]
```

```
      },
      "Action": [
        "es:ESHttp*"
      ],
      "Resource": [
        "arn:aws:es:us-west-1:987654321098:domain/test-domain/test-index/*",
        "arn:aws:es:us-west-1:987654321098:domain/test-domain/_bulk"
      ]
    },
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "arn:aws:iam::123456789012:user/test-user"
        ]
      },
      "Action": [
        "es:ESHttpGet"
      ],
      "Resource": "arn:aws:es:us-west-1:987654321098:domain/test-domain/
restricted-index/*"
    }
  ]
}
```

Esta política otorga a `test-user` acceso completo a `test-index` y la API bulk de OpenSearch. También permite solicitudes `GET` a `restricted-index`.

La siguiente solicitud de indexación, como cabría esperar, falla debido a un error de permisos:

```
PUT https://search-test-domain.us-west-1.es.amazonaws.com/restricted-index/movie/1
{
  "title": "Your Name",
  "director": "Makoto Shinkai",
  "year": "2016"
}
```

A diferencia de la API index, la API bulk permite crear, actualizar y eliminar muchos documentos en una única llamada. Sin embargo, con frecuencia especifica estas operaciones en el cuerpo de la solicitud, en lugar de la URL de solicitud. Como OpenSearch Service utiliza direcciones URL para controlar el acceso a subrecursos de dominio, `test-user` puede, de hecho, utilizar la API bulk para

realizar cambios en `restricted-index`. Aunque el usuario carezca de permisos `POST` en el índice, la siguiente solicitud tiene éxito:

```
POST https://search-test-domain.us-west-1.es.amazonaws.com/_bulk
{ "index" : { "_index": "restricted-index", "_type" : "movie", "_id" : "1" } }
{ "title": "Your Name", "director": "Makoto Shinkai", "year": "2016" }
```

En esta situación, la política de acceso no consigue cumplir su objetivo. Para evitar que los usuarios omitan este tipo de restricciones, puede cambiar `rest.action.multi.allow_explicit_index` a falso. Si este valor es falso, todas las llamadas a las API bulk, mget y msearch que especifican nombres de índice en el cuerpo de la solicitud dejan de funcionar. En otras palabras, las llamadas a `_bulk` ya no funcionan, pero las llamadas a `test-index/_bulk` sí. Este segundo punto de conexión contiene un nombre de índice, por lo que no es necesario especificar uno en el cuerpo de la solicitud.

OpenSearch Dashboards depende en gran medida de mget y msearch, por lo que es improbable trabajar correctamente después de este cambio. Como remedio parcial, puede dejar `rest.action.multi.allow_explicit_index` como verdadero y denegar a determinados usuarios el acceso a una o varias de estas API.

Para obtener más información acerca de cómo cambiar esta configuración, consulte the section called “Configuración avanzada de clústeres”.

Del mismo modo, la siguiente política basada en recursos contiene dos problemas sutiles:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::123456789012:user/test-user"
      },
      "Action": "es:ESHttp*",
      "Resource": "arn:aws:es:us-west-1:987654321098:domain/test-domain/*"
    },
    {
      "Effect": "Deny",
```

```
      "Principal": {
        "AWS": "arn:aws:iam::123456789012:user/test-user"
      },
      "Action": "es:ESHttp*",
      "Resource": "arn:aws:es:us-west-1:987654321098:domain/test-domain/
restricted-index/*"
    }
  ]
}
```

- A pesar de la denegación explícita, `test-user` puede seguir realizando llamadas como, por ejemplo, `GET https://search-test-domain.us-west-1.es.amazonaws.com/_all/_search` y `GET https://search-test-domain.us-west-1.es.amazonaws.com/*/_search` para acceder a los documentos en `restricted-index`.
- Dado que el elemento `Resource` hace referencia a `restricted-index/*`, `test-user` no tiene permisos para acceder directamente a los documentos del índice. El usuario, sin embargo, tiene permisos para eliminar todo el índice. Para evitar el acceso y la eliminación, la política debe especificar en su lugar `restricted-index*`.

En lugar de combinar permisos amplios y denegaciones delimitadas, el enfoque más seguro consiste en seguir el principio de privilegio mínimo y conceder solo los permisos requeridos para realizar una tarea. Para obtener más información sobre cómo controlar el acceso a índices u operaciones de OpenSearch individuales, consulte the section called “Control de acceso detallado”.

> **Important**
>
> Especificar el comodín * permite el acceso anónimo al dominio. No se recomienda el uso del comodín. Además, analice detenidamente las siguientes políticas para confirmar que no otorgan acceso amplio:
>
> - Políticas basadas en identidad asociadas a las entidades principales de AWS vinculadas (por ejemplo, los roles de IAM)
> - Políticas basadas en recursos adjuntas a recursos de AWS asociados (por ejemplo, claves de AWS Key Management Service [KMS])

Configurar políticas de acceso

- Para obtener instrucciones sobre la creación o modificación de políticas basadas en recursos y en IP en OpenSearch Service, consulte the section called “Configuración de políticas de acceso”.
- Para obtener instrucciones acerca de la creación o modificación de políticas basadas en identidad en IAM, consulte Definición de permisos de IAM personalizados con políticas administradas por el cliente en la Guía del usuario de IAM.

Políticas de muestra adicionales

Aunque este capítulo incluye muchas políticas de muestra, el control de acceso de AWS es un tema complejo que se entiende mejor a través de ejemplos. Para más información, consulte Ejemplos de políticas de IAM basadas en identidad en la Guía del usuario de IAM.

Referencia de permisos de la API de Amazon OpenSearch Service

Cuando configura el control de acceso, escribe políticas de permiso que puede adjuntar a una identidad de IAM (políticas basadas en identidad). Para obtener más información detallada de referencia, consulte los siguientes temas en la Referencia de autorizaciones de servicio:

- Acciones, recursos y claves de condición para OpenSearch Service.
- Acciones, recursos y claves de condición para OpenSearch Ingestion.

Esta referencia contiene información sobre las operaciones de la API que se pueden utilizar en una política de IAM. También incluye el recurso de AWS para el cual puede conceder permisos y claves de condición que puede incluir para un control de acceso detallado.

Las acciones se especifican en el campo `Action` de la política, el valor del recurso en el campo `Resource` de la política y las condiciones en el campo `Condition` de la política. Para especificar una acción para OpenSearch Service, use el prefijo `es:` seguido del nombre de operación de la API (por ejemplo, `es:CreateDomain`). Para especificar una acción para OpenSearch Ingestion, use el prefijo `osis:` seguido del nombre de operación de la API (por ejemplo, `osis:CreatePipeline`).

Políticas administradas de AWS para Amazon OpenSearch Service

Una política administrada de AWS es una política independiente que AWS crea y administra. Las políticas administradas de AWS se diseñan para ofrecer permisos para muchos casos de uso comunes, por lo que puede empezar a asignar permisos a los usuarios, grupos y roles.

Considere que es posible que las políticas administradas de AWS no concedan permisos de privilegio mínimo para los casos de uso concretos, ya que están disponibles para que las utilicen todos los clientes de AWS. Se recomienda definir políticas administradas por el cliente específicas para sus casos de uso a fin de reducir aún más los permisos.

No puede cambiar los permisos definidos en las políticas administradas de AWS. Si AWS actualiza los permisos definidos en una política administrada de AWS, la actualización afecta a todas las identidades de entidades principales (usuarios, grupos y roles) a las que está adjunta la política. Lo más probable es que AWS actualice una política administrada de AWS cuando se lance un nuevo Servicio de AWS o las operaciones de la API nuevas estén disponibles para los servicios existentes.

Para obtener más información, consulte Políticas administradas de AWS en la Guía del usuario de IAM.

AmazonOpenSearchDirectQueryGlueCreateAccess

Concede al servicio de consultas directas de Amazon OpenSearch Service acceso a `CreateDatabase`, `CreatePartition`, `CreateTable` y `BatchCreatePartition` AWS Glue API.

Puede encontrar la política AmazonOpenSearchDirectQueryGlueCreateAccess en la consola de IAM.

AmazonOpenSearchServiceFullAccess

Concede acceso completo a todos los recursos y operaciones de la API de configuración de OpenSearch Service para una Cuenta de AWS.

Puede encontrar la política AmazonOpenSearchServiceFullAccess en la consola de IAM.

AmazonOpenSearchServiceReadOnlyAccess

Concede acceso de solo lectura a todos los recursos de OpenSearch Service para una Cuenta de AWS.

Puede encontrar la política AmazonOpenSearchServiceReadOnlyAccess en la consola de IAM.

AmazonOpenSearchServiceRolePolicy

No puede adjuntar `AmazonOpenSearchServiceRolePolicy` a sus entidades de IAM. Esta política se adjunta a un rol vinculado a servicio que permite a OpenSearch Service acceder a los recursos de la cuenta. Para más información, consulte the section called “Permisos”.

Puede encontrar la política AmazonOpenSearchServiceRolePolicy en la consola de IAM.

AmazonOpenSearchServiceCognitoAccess

Proporciona los permisos mínimos de Amazon Cognito necesarios para habilitar la autenticación de Cognito.

Puede encontrar la política AmazonOpenSearchServiceCognitoAccess en la consola de IAM.

AmazonOpenSearchIngestionServiceRolePolicy

No puede asociar `AmazonOpenSearchIngestionServiceRolePolicy` a sus entidades IAM. Esta política se adjunta a un rol vinculado a servicios que permite a OpenSearch Ingestion habilitar el acceso a VPC para canalizaciones de ingestión, crear etiquetas y publicar métricas de CloudWatch relacionadas con la incorporación en su cuenta. Para más información, consulte the section called “Cómo utilizar roles vinculados a servicios”.

Puede encontrar la política AmazonOpenSearchIngestionServiceRolePolicy en la consola de IAM.

OpenSearchIngestionSelfManagedVpcePolicy

No puede asociar `OpenSearchIngestionSelfManagedVpcePolicy` a sus entidades IAM. Esta política se adjunta a un rol vinculado a servicios que permite a OpenSearch Ingestion habilitar el acceso a VPC autogestionadas para canalizaciones de ingesta, crear etiquetas y publicar métricas de CloudWatch relacionadas con la ingesta en su cuenta. Para obtener más información, consulte the section called “Cómo utilizar roles vinculados a servicios”.

Puede encontrar la política OpenSearchIngestionSelfManagedVpcePolicy en la consola de IAM.

AmazonOpenSearchIngestionFullAccess

Concede acceso completo a las operaciones y recursos de la API de OpenSearch Ingestion para una Cuenta de AWS.

Puede encontrar la política AmazonOpenSearchIngestionFullAccess en la consola de IAM.

Acceso de solo lectura a Amazon OpenSearchingestion

Concede acceso de solo lectura a todos los recursos de OpenSearch Ingestion para una Cuenta de AWS.

Puede encontrar la política AmazonOpenSearchIngestionReadOnlyAccess en la consola de IAM.

AmazonOpenSearchServerlessServiceRolePolicy

Proporciona los permisos de Amazon CloudWatch mínimos necesarios para enviar datos de métricas de OpenSearch sin servidor a CloudWatch.

Puede encontrar la política AmazonOpenSearchServerlessServiceRolePolicy en la consola de IAM.

Actualizaciones de OpenSearch Service de las políticas administradas de AWS

Es posible consultar los detalles sobre las actualizaciones de las políticas administradas de AWS para OpenSearch Service debido a que este servicio comenzó a realizar el seguimiento de estos cambios.

Cambio	Descripción	Fecha
Se han actualizado los `AmazonOpenSearchIngestionServiceRolePolicy`	La actualización otorga a OpenSearch Ingestion permiso para modificar los puntos de conexión de VPC creados por OpenSearch para compartir canalizaciones entre las VPC. Para ver la política JSON, consulte la consola de IAM.	28 de agosto de 2025
Se han actualizado los `AmazonOpenSearchServiceRolePolicy`	Se agregó la siguiente sentencia a la política. Cuando Amazon OpenSearch Service asume el rol vinculado al servicio `AWSServiceRoleForAmazonOpenSearchSer`	31 de marzo de 2025

Cambio	Descripción	Fecha
	`vice` , esta nueva sentencia de la política permite a OpenSearch actualizar el alcance de acceso de cualquier aplicación de AWS IAM Identity Center que solo esté administrada por OpenSearch. `{ "Effect": "Allow", "Action": "sso:PutApplicationAccessScope", "Resource": "arn:aws:sso::*:application/*/*", "Condition": { "StringEquals": { "aws:ResourceOrgID": "${aws:PrincipalOrgID}" } } }`	
Se ha actualizado `AmazonOpenSearchServerlessServiceRolePolicy`	Se agregó el Sid `AllowAOSSCloudwatchMetrics` a la política `AmazonOpenSearchServerlessServiceRolePolicy` . Un Sid es un ID de instrucción que actúa como identificador opcional para la instrucción de la política.	12 de julio de 2024

Cambio	Descripción	Fecha
Se ha agregado `OpenSearchIngestionSelfManagedVpcePolicy`	Una nueva política que permite a OpenSearch Ingestion habilitar el acceso a VPC autogestionadas para canalizaciones de ingesta, crear etiquetas y publicar métricas de CloudWatch relacionadas con la ingesta en su cuenta. Para ver la política JSON, consulte la consola de IAM.	12 de junio de 2024
Added`AmazonOpenSearchDirectQueryGlueCreateAccess`	Concede al servicio de consultas directas de Amazon OpenSearch Service acceso a `CreateDatabase` , `CreatePartition` , `CreateTable` y `BatchCreatePartition` AWS Glue API.	6 de mayo de 2024
Actualización de `AmazonOpenSearchServiceRolePolicy` y `AmazonElasticsearchServiceRolePolicy`	Se añadieron los permisos necesarios para que el rol vinculado al servicio asigne y desasigne direcciones IPv6. La política de Elasticsearch en desuso también se ha actualizado para garantizar la compatibilidad con versiones anteriores.	18 de octubre de 2023

Cambio	Descripción	Fecha
Se ha agregado `AmazonOpenSearchIngestionServiceRolePolicy`	Una nueva política que permite a OpenSearch Ingestion habilitar el acceso a VPC para canalizaciones de ingestión, crear etiquetas y publicar métricas de CloudWatch relacionadas con la incorporación en su cuenta. Para ver la política JSON, consulte la consola de IAM.	26 de abril de 2023
Se ha agregado `AmazonOpenSearchIngestionFullAccess`	Una nueva política que concede acceso completo a las operaciones y recursos de la API de OpenSearch Ingestion para una Cuenta de AWS. Para ver la política JSON, consulte la consola de IAM.	26 de abril de 2023
Se ha agregado `AmazonOpenSearchIngestionReadOnlyAccess`	Una nueva política que concede acceso de solo lectura a todos los recursos de OpenSearch Ingestion para una Cuenta de AWS. Para ver la política JSON, consulte la consola de IAM.	26 de abril de 2023

Cambio	Descripción	Fecha
Se ha agregado `AmazonOpenSearchServerlessServiceRolePolicy`	Una nueva política que proporciona los permisos mínimos necesarios para enviar datos de métricas de OpenSearch sin servidor a Amazon CloudWatch. Para ver la política JSON, consulte la consola de IAM.	29 de noviembre de 2022
Actualización de `AmazonOpenSearchServiceRolePolicy` y `AmazonElasticsearchServiceRolePolicy`	Se agregaron los permisos necesarios para que el rol vinculado al servicio cree puntos de conexión de VPC administrados por OpenSearch Service. Algunas acciones solo se pueden realizar cuando la solicitud contiene la etiqueta `OpenSearchManaged=true` . La política de Elasticsearch en desuso también se ha actualizado para garantizar la compatibilidad con versiones anteriores.	7 de noviembre de 2022

Cambio	Descripción	Fecha
Actualización de `AmazonOpenSearchServiceRolePolicy` y `AmazonElasticsearchServiceRolePolicy`	Se ha agregado compatibilidad para la acción `PutMetricData`, que es necesaria para publicar métricas de clústeres de OpenSearch en Amazon CloudWatch. La política de Elasticsearch en desuso también se ha actualizado para garantizar la compatibilidad con versiones anteriores. Para ver la política JSON, consulte la consola de IAM.	12 de septiembre de 2022

Cambio	Descripción	Fecha
Actualización de `AmazonOpenSearchServiceRolePolicy` y `AmazonElasticsearchServiceRolePolicy`	Se ha agregado compatibilidad con el tipo de recurso `acm`. La política proporciona el permiso mínimo de solo lectura AWS Certificate Manager (ACM) necesario para el rol vinculado al servicio para verificar y validar los recursos de ACM a fin de crear y actualizar los dominios habilitados para el punto de conexión personalizado. La política de Elasticsearch en desuso también se ha actualizado para garantizar la compatibilidad con versiones anteriores.	28 de julio de 2022

Cambio	Descripción	Fecha
Actualización de `AmazonOpenSearchServiceCognitoAccess` y `AmazonESCognitoAccess`	Se ha agregado compatibilidad con la acción `UpdateUserPoolClient`, que es necesaria para establecer la configuración del grupo de usuarios de Cognito durante la actualización de Elasticsearch a OpenSearch. Se han corregido los permisos de la acción `SetIdentityPoolRoles` para permitir el acceso a todos los recursos. La política de Elasticsearch en desuso también se ha actualizado para garantizar la compatibilidad con versiones anteriores.	20 de diciembre de 2021
Se ha actualizado `AmazonOpenSearchServiceRolePolicy`	Se ha agregado compatibilidad con el tipo de recurso `security-group`. La política proporciona los permisos mínimos de Amazon EC2 y Elastic Load Balancing necesarios para el rol vinculado a servicios a fin de habilitar el acceso mediante la VPC.	9 de septiembre de 2021

Cambio	Descripción	Fecha
• Se ha agregado `AmazonOpenSearchServiceFullAccess` • Se ha dado de baja `AmazonESFullAccess`	Esta nueva política tiene por objeto sustituir a la política anterior. Ambas políticas proporcionan acceso completo a la API de configuración de OpenSearch Service y a todos los métodos HTTP para las API de OpenSearch. El control de acceso detallado y las políticas basadas en recursos aún pueden restringir el acceso.	7 de septiembre de 2021
• Se ha agregado `AmazonOpenSearchServiceReadOnlyAccess` • Se ha dado de baja `AmazonESReadOnlyAccess`	Esta nueva política tiene por objeto sustituir a la política anterior. Ambas políticas proporcionan acceso de solo lectura a la API de configuración de OpenSearch Service (`es:Describe*`, `es:List*` y `es:Get*`) y no brindan acceso a los métodos HTTP para las API de OpenSearch.	7 de septiembre de 2021
• Se ha agregado `AmazonOpenSearchServiceCognitoAccess` • Se ha dado de baja `AmazonESCognitoAccess`	Esta nueva política tiene por objeto sustituir a la política anterior. Ambas políticas proporcionan los permisos mínimos de Amazon Cognito necesarios para habilitar la autenticación de Cognito.	7 de septiembre de 2021

Cambio	Descripción	Fecha
• Se ha agregado AmazonOpenSearchServiceRolePolicy • Se ha dado de baja `AmazonElasticsearchServiceRolePolicy`	Esta nueva política tiene por objeto sustituir a la política anterior. Ambas políticas proporcionan los permisos mínimos de Amazon EC2 y Elastic Load Balancing necesarios para el rol vinculado a servicios a fin de habilitar el acceso mediante la VPC.	7 de septiembre de 2021
Comenzó el seguimiento de los cambios	Amazon OpenSearch Service ahora realiza un seguimiento de los cambios en las políticas administradas de AWS.	7 de septiembre de 2021

Prevención de la sustitución confusa entre servicios

El problema de la sustitución confusa es un problema de seguridad en el que una entidad que no tiene permiso para realizar una acción puede obligar a una entidad con más privilegios a realizar la acción. En AWS, la suplantación entre servicios puede dar lugar al problema de la sustitución confusa. La suplantación entre servicios puede producirse cuando un servicio (el servicio que lleva a cabo las llamadas) llama a otro servicio (el servicio al que se llama). El servicio que lleva a cabo las llamadas se puedes manipular para utilizar sus permisos a fin de actuar en función de los recursos de otro cliente de una manera en la que no debe tener permiso para acceder. Para evitarlo, AWS proporciona herramientas que lo ayudan a proteger sus datos para todos los servicios con entidades principales de servicio a las que se les ha dado acceso a los recursos de su cuenta.

Se recomienda utilizar las claves de contexto de condición global `aws:SourceArn` y `aws:SourceAccount` en las políticas de recursos para limitar los permisos que Amazon OpenSearch Service concede a otro servicio para el recurso. Si el valor de `aws:SourceArn` no contiene el ID de cuenta, como un ARN de bucket de Amazon S3, debe utilizar ambas claves de contexto de condición global para limitar los permisos. Si utiliza claves de contexto de condición

global y el valor de `aws:SourceArn` contiene el ID de cuenta, el valor de `aws:SourceAccount` y la cuenta en el valor de `aws:SourceArn` deben utilizar el mismo ID de cuenta cuando se utiliza en la misma instrucción de política. Utiliza `aws:SourceArn` si desea que solo se asocie un recurso al acceso entre servicios. Utiliza `aws:SourceAccount` si quiere permitir que cualquier recurso de esa cuenta se asocie al uso entre servicios.

El valor de `aws:SourceArn` debe ser el ARN del dominio OpenSearch Service.

La forma más eficaz de protegerse contra el problema de la sustitución confusa es utilizar la clave de contexto de condición global de `aws:SourceArn` con el ARN completo del recurso. Si no conoce el ARN completo del recurso o si especifica varios recursos, utiliza la clave de condición de contexto global `aws:SourceArn` con comodines (*) para las partes desconocidas del ARN. Por ejemplo, `arn:aws:es:*:123456789012:*`.

El siguiente ejemplo muestra cómo se pueden utilizar las claves contextuales de condición global `aws:SourceArn` y `aws:SourceAccount` en OpenSearch Service para evitar el problema del adjunto confundido.

JSON

```
{
   "Version":"2012-10-17",
   "Statement":{
      "Sid":"ConfusedDeputyPreventionExamplePolicy",
      "Effect":"Allow",
      "Principal":{
         "Service":"es.amazonaws.com"
      },
      "Action":"sts:AssumeRole",
      "Condition":{
         "StringEquals":{
            "aws:SourceAccount":"111122223333"
         },
         "ArnLike":{
            "aws:SourceArn":"arn:aws:es:us-east-1:111122223333:domain/my-domain"
         }
      }
   }
}
```

Control de acceso detallado en Amazon OpenSearch Service

El control de acceso detallado ofrece formas adicionales de controlar el acceso a los datos en Amazon OpenSearch Service. Por ejemplo, puede ser conveniente que, según quién realice la solicitud, una búsqueda devuelva los resultados de un solo índice. Es posible que desee ocultar determinados campos en los documentos o excluir algunos documentos por completo.

El control de acceso detallado ofrece las siguientes características:

- Control de acceso con base en roles
- Seguridad en el nivel de índice, de documento y de campo
- Varios inquilinos en OpenSearch Dashboards
- Autenticación básica de HTTP para OpenSearch y OpenSearch Dashboards

Temas

- Panorámica general: control de acceso detallado y seguridad de OpenSearch Service
- Conceptos clave
- Acerca del usuario maestro
- Habilitar el control de acceso detallado
- Acceder a OpenSearch Dashboards como usuario maestro
- Administrar permisos
- Configuraciones recomendadas
- Limitaciones
- Modificar el usuario maestro
- Usuarios maestros adicionales
- Instantáneas manuales
- Integraciones
- Diferencias en la API de REST
- Tutorial: Configurar un dominio con un usuario maestro de IAM y la autenticación de Amazon Cognito
- Tutorial: Configurar un dominio con la base de datos de usuarios interna y la autenticación básica de HTTP

Panorámica general: control de acceso detallado y seguridad de OpenSearch Service

La seguridad de Amazon OpenSearch Service tiene tres capas principales:

Network

La primera capa de seguridad es la red, que determina si las solicitudes llegarán a un dominio de OpenSearch Service. Si elige Acceso público al crear un dominio, las solicitudes procedentes de cualquier cliente conectado a Internet pueden llegar al punto de conexión del dominio. Si elige Acceso a la VPC, los clientes deben conectarse a la VPC (y los grupos de seguridad asociados deben permitirlo) para que una solicitud llegue al punto de conexión. Para obtener más información, consulte the section called “Compatibilidad con VPC”.

Política de acceso al dominio

La segunda capa de seguridad es la política de acceso al dominio. Una vez que una solicitud llega a un punto de conexión del dominio, la política de acceso con base en recursos permite o deniega el acceso de la solicitud a un URI determinado. La política de acceso acepta o rechaza las solicitudes en el “límite” del dominio, antes de que lleguen a OpenSearch propiamente dicho.

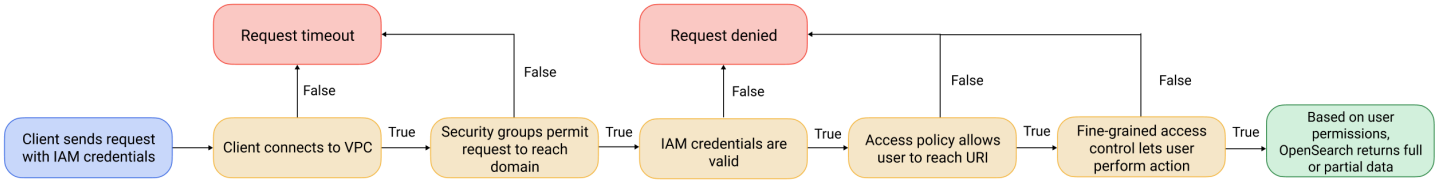
Control de acceso detallado

La tercera y última capa de seguridad es el control de acceso detallado. Una vez que una política de acceso con base en recursos ha permitido que una solicitud llegue a un punto de conexión del dominio, el control de acceso detallado evalúa las credenciales del usuario y lo autentica o bien deniega la solicitud. Si el control de acceso detallado autentica al usuario, obtiene todos los roles mapeados a ese usuario y utiliza el conjunto completo de permisos para determinar cómo gestionar la solicitud.

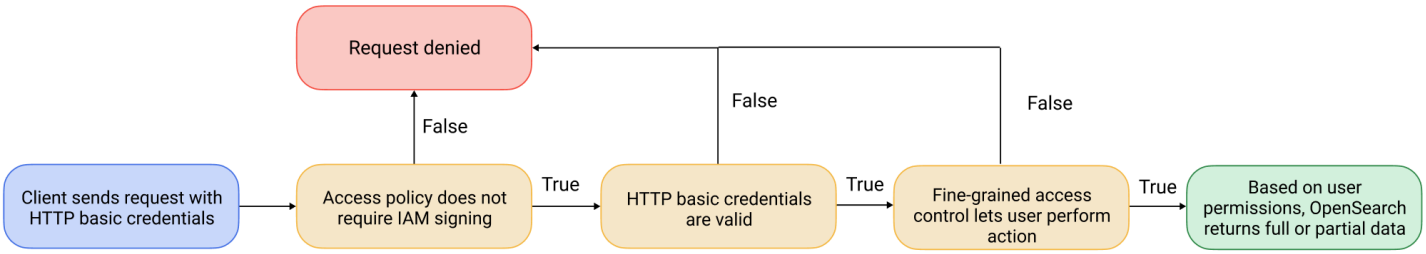
> **Note**
>
> Si una política de acceso basada en recursos contiene usuarios o roles de IAM, los clientes deben enviar solicitudes firmadas mediante AWS Signature Version 4. Como tales, las políticas de acceso pueden entrar en conflicto con el control de acceso detallado, sobre todo si se utiliza la base de datos de usuarios interna y la autenticación básica de HTTP. No se puede firmar una solicitud con un nombre de usuario y contraseña y también con

credenciales de IAM. En general, si habilita el control de acceso detallado, se recomienda utilizar una política de acceso al dominio que no requiera solicitudes firmadas.

Este primer diagrama ilustra una configuración habitual: un dominio de acceso a la VPC con el control de acceso detallado habilitado, una política de acceso con base en IAM y un usuario maestro de IAM.



Este segundo diagrama ilustra otra configuración habitual: un dominio de acceso público con el control de acceso detallado habilitado, una política de acceso que no utiliza principales de IAM y un usuario maestro en la base de datos de usuarios interna.



Ejemplo

Supongamos que se realiza una solicitud `GET` a `movies/_search?q=thor`. ¿El usuario tiene permisos para buscar en el índice `movies`? En caso afirmativo, ¿el usuario tiene permisos para ver todos los documentos que contiene? ¿La respuesta debería omitir o anonimizar algún campo? Para el usuario maestro, la respuesta podría tener este aspecto:

```
{
  "hits": {
    "total": 7,
    "max_score": 8.772789,
    "hits": [{
        "_index": "movies",
        "_type": "_doc",
        "_id": "tt0800369",
        "_score": 8.772789,
        "_source": {
```

```
          "directors": [
            "Kenneth Branagh",
            "Joss Whedon"
          ],
          "release_date": "2011-04-21T00:00:00Z",
          "genres": [
            "Action",
            "Adventure",
            "Fantasy"
          ],
          "plot": "The powerful but arrogant god Thor is cast out of Asgard to
 live amongst humans in Midgard (Earth), where he soon becomes one of their finest
 defenders.",
          "title": "Thor",
          "actors": [
            "Chris Hemsworth",
            "Anthony Hopkins",
            "Natalie Portman"
          ],
          "year": 2011
        }
      },
      ...
    ]
  }
}
```

Si un usuario con permisos más limitados emite exactamente la misma solicitud, la respuesta podría tener el siguiente aspecto:

```
{
  "hits": {
    "total": 2,
    "max_score": 8.772789,
    "hits": [{
        "_index": "movies",
        "_type": "_doc",
        "_id": "tt0800369",
        "_score": 8.772789,
        "_source": {
          "year": 2011,
          "release_date":
 "3812a72c6dd23eef3c750c2d99e205cbd260389461e19d610406847397ecb357",
```

```
          "plot": "The powerful but arrogant god Thor is cast out of Asgard to
 live amongst humans in Midgard (Earth), where he soon becomes one of their finest
 defenders.",
          "title": "Thor"
        }
      },
      ...
    ]
  }
}
```

La respuesta contiene menos aciertos y menos campos para cada acierto. Además, el campo `release_date` está anonimizado. Si un usuario sin permisos realiza la misma solicitud, el clúster devuelve un error:

```
{
  "error": {
    "root_cause": [{
      "type": "security_exception",
      "reason": "no permissions for [indices:data/read/search] and User [name=limited-
user, roles=[], requestedTenant=null]"
    }],
    "type": "security_exception",
    "reason": "no permissions for [indices:data/read/search] and User [name=limited-
user, roles=[], requestedTenant=null]"
  },
  "status": 403
}
```

Si un usuario proporciona credenciales no válidas, el clúster devuelve una excepción `Unauthorized`.

Conceptos clave

Al comenzar con el control de acceso detallado, tenga en cuenta los siguientes conceptos:

- Roles: la forma principal de utilizar el control de acceso detallado. En este caso, se trata de roles distintos de los roles de IAM. Los roles contienen cualquier combinación de permisos: para todo el clúster, específicos del índice, en el nivel de documento y en el nivel de campo.
- Asignación: después de configurar un rol, se asigna a uno o más usuarios. Por ejemplo, puede mapear tres roles a un solo usuario: un rol que proporciona acceso a Dashboards, otro que

proporciona acceso de solo lectura a `index1` y otro que proporciona acceso de escritura a `index2`. Si lo prefiere, puede incluir todos esos permisos en un solo rol.

- Usuarios: personas o aplicaciones que hacen solicitudes al clúster de OpenSearch. Los usuarios tienen credenciales, ya sean claves de acceso de IAM o un nombre de usuario y una contraseña, que especifican al realizar las solicitudes.

Acerca del usuario maestro

El usuario maestro de OpenSearch Service es una combinación de nombre de usuario y contraseña, o una entidad principal de IAM, que tiene todos los permisos para acceder al clúster de OpenSearch subyacente. Se considera que un usuario es un usuario maestro si tiene todos los accesos al clúster de OpenSearch y la capacidad de crear usuarios internos, roles y asignaciones de roles en OpenSearch Dashboards.

Un usuario maestro creado en la consola de OpenSearch Service o mediante la CLI se asigna automáticamente a dos funciones predefinidas:

- `all_access`: proporciona acceso total a todas las operaciones del clúster, permiso para escribir en todos los índices del clúster y permiso para escribir en todos los inquilinos.
- `security_manager`: proporciona acceso al complemento de seguridad y administración de los usuarios y los permisos.

Con estos dos roles, el usuario accede a la pestaña Seguridad en OpenSearch Dashboards, donde puede gestionar los usuarios y los permisos. Si crea otro usuario interno y solo lo asigna al rol `all_access`, el usuario no tendrá acceso a la pestaña Seguridad. Para crear usuarios maestros adicionales, puede asignarlos de forma explícita a los roles `all_access` y `security_manager`. Para obtener instrucciones, consulte the section called “Usuarios maestros adicionales”.

Al crear un usuario maestro para su dominio, puede especificar una entidad principal de IAM existente o crear un usuario maestro en la base de datos de usuarios interna. Tenga en cuenta lo siguiente cuando tenga que tomar una decisión sobre el uso:

- Entidad principal de IAM: si elige IAM para el usuario maestro, todas las solicitudes al clúster deben firmarse mediante AWS Signature Version 4.

  OpenSearch Service no tiene en cuenta ninguno de los permisos de la entidad principal de IAM. El rol o usuario de IAM sirve únicamente para la autenticación. Las políticas de ese usuario o rol

no influyen en la autorización del usuario maestro. La autorización se gestiona a través de los distintos permisos del complemento OpenSearch Security.

Por ejemplo, puede asignar cero permisos de IAM a una entidad principal de IAM y, siempre que la máquina o la persona puedan autenticarse ante ese usuario o rol, tendrán el poder del usuario maestro en OpenSearch Service.

Recomendamos IAM si desea utilizar los mismos usuarios en varios clústeres, si desea utilizar Amazon Cognito para obtener acceso a los Dashboards o si tiene clientes de OpenSearch que admiten la firma con Signature Version 4.

- Base de datos de usuarios interna: si crea una base de datos maestra en la base de datos de usuarios interna (con una combinación de nombre de usuario y contraseña), puede utilizar la autenticación básica HTTP (además de las credenciales de IAM) para hacer solicitudes al clúster. La mayoría de los clientes admiten la autenticación básica, incluido curl, que también admite AWS Signature Version 4 con--aws-sigv4 option. La base de datos de usuarios interna se almacena en un índice de OpenSearch, por lo que no puede compartirla con otros clústeres.

  Recomendamos la base de datos de usuarios interna si no se requiere reutilizar los usuarios en varios clústeres, si se desea utilizar la autenticación básica de HTTP para obtener acceso a Dashboards (en lugar de Amazon Cognito) o si se tienen clientes que solo admiten la autenticación básica. La base de datos de usuarios interna es la forma más sencilla de comenzar a utilizar OpenSearch Service.

Habilitar el control de acceso detallado

Habilite el control de acceso detallado mediante la consola, la AWS CLI o la API de configuración. Para ver los pasos, consulte Creación y administración de dominios.

El control de acceso detallado requiere OpenSearch o Elasticsearch 6.7 posteriores. También requiere HTTPS para todo el tráfico al dominio, Cifrado de datos en reposo, y cifrado de nodo a nodo. Según cómo configure las características avanzadas del control de acceso detallado, el procesamiento adicional de sus solicitudes puede requerir recursos de cómputo y memoria en nodos de datos individuales. Después de habilitar el control de acceso detallado, no puede desactivarlo.

Habilitar el control de acceso detallado en los dominios existentes

Puede habilitar el control de acceso detallado en dominios existentes que ejecutan OpenSearch o Elasticsearch 6.7 o posteriores.

Para habilitar el control de acceso detallado en un dominio existente (consola)

1. Seleccione el dominio y elija Acciones y Editar la configuración de seguridad.
2. Seleccione Habilitar el control de acceso detallado,
3. Seleccione cómo crear el usuario maestro:

   - Si desea utilizar IAM para la gestión de usuarios, seleccione Establecer ARN de IAM como usuario maestro y especifique el ARN para un rol de IAM.
   - Si desea utilizar la base de datos de usuarios interna, seleccione Crear usuario maestro y especifique un nombre de usuario y una contraseña.
4. (Opcional) Seleccione Habilitar el período de migración para la política de acceso abierta o basada en IP. Esta configuración permite un período de transición de 30 días durante el cual los usuarios existentes pueden seguir accediendo al dominio sin interrupciones, y las políticas de acceso basadas en IP y abiertas existentes seguirán funcionando con su dominio. Durante este período de migración, recomendamos a los administradores crear los roles necesarios y asignarlos a los usuarios para el dominio. Si utiliza políticas basadas en la identidad en lugar de una política de acceso abierta o basada en la IP, puede desactivar esta configuración.

   También debe actualizar sus clientes para que trabajen con un control de acceso de grano fino durante el periodo de migración. Por ejemplo, si asigna roles de IAM con control de acceso detallado, debe actualizar los clientes para que empiecen a firmar las solicitudes con AWS Signature Version 4. Si configura la autenticación básica HTTP con un control de acceso detallado, debe actualizar los clientes para proporcionar las credenciales de autenticación básicas adecuadas en las solicitudes.

   Durante el período de migración, los usuarios que acceden al punto de conexión de OpenSearch Dashboards del dominio aterrizarán directamente en la página Descubrir en lugar de la página de inicio de sesión. Los administradores y los usuarios maestros pueden elegir Inicio de sesión para iniciar sesión con credenciales de administrador y configurar asignaciones de roles.

   > **Important**
   >
   > OpenSearch Service desactiva automáticamente el período de migración transcurridos 30 días. Recomendamos finalizarlo tan pronto como cree los roles necesarios y los asigne a los usuarios. Una vez finalizado el período de migración, no puedes volver a habilitarlo.

5. Seleccione Guardar cambios.

El cambio desencadena una implementación azul-verde durante el cual el estado del clúster se vuelve rojo, pero todas las operaciones de clúster no se ven afectadas.

Para habilitar el control de acceso detallado en un dominio (CLI) existente)

Establezca `AnonymousAuthEnabled` a `true` para habilitar el período de migración con un control de acceso detallado:

```
aws opensearch update-domain-config --domain-name test-domain --region us-east-1 \
      --advanced-security-options '{ "Enabled": true,
 "InternalUserDatabaseEnabled":true, "MasterUserOptions": {"MasterUserName":"master-
username","MasterUserPassword":"master-password"},"AnonymousAuthEnabled": true}'
```

Acerca del default_role

El control de acceso detallado requiere asignación de roles. Si el dominio utiliza políticas de acceso basadas en identidades, OpenSearch Service asigna automáticamente a los usuarios de IAM a un nuevo rol denominado default_role para ayudar a migrar correctamente a los usuarios existentes. Esta asignación temporal garantiza que los usuarios puedan seguir enviando correctamente solicitudes GET y PUT firmadas por IAM hasta que cree sus propias asignaciones de roles.

El rol no agrega vulnerabilidades ni fallos de seguridad a su dominio de OpenSearch Service. Recomendamos eliminar el rol predeterminado tan pronto como configure sus propios roles y los asigne en consecuencia.

Escenarios de migración

La siguiente tabla describe el comportamiento de cada método de autenticación antes y después de habilitar el control de acceso detallado en un dominio existente, y los pasos que deben seguir los administradores para asignar correctamente sus usuarios a los roles:

Método de autenticación	Antes de habilitar el control de acceso detallado	Después de habilitar el control de acceso detallado	Tareas administrativas
Políticas basadas en identidades	Todos los usuarios que cumplan la política de IAM pueden acceder al dominio.	No es necesario habilitar el período de migración. OpenSearch Service asigna automáticamente a todos los usuarios de IAM que cumplen la política de IAM a default_role para que puedan seguir accediendo al dominio.	1. Cree asignaciones de roles personalizadas en el dominio. 2. Elimine las default_role.
Políticas basadas en IP	Todos los usuarios de las direcciones IP permitidas o los bloques de CIDR pueden acceder al dominio.	Durante el período de migración de 30 días, todos los usuarios de las direcciones IP permitidas o los bloques de CIDR pueden seguir accediendo al dominio.	1. Cree asignaciones de roles personalizadas en el dominio. 2. Actualice sus clientes para proporcionar credenciales de autenticación básicas o credenciales de IAM, según la configuración de asignación de roles. 3. Desactive el período de migración. Los usuarios de las direcciones IP permitidas o los bloques de CIDR que envían solicitudes sin autenticación básica o credenciales de IAM perderán el acceso al dominio.
Políticas de acceso abierto	Todos los usuarios en Internet	Durante el periodo de migración de 30 días, todos los usuarios	1. Cree asignaciones de roles en el dominio.

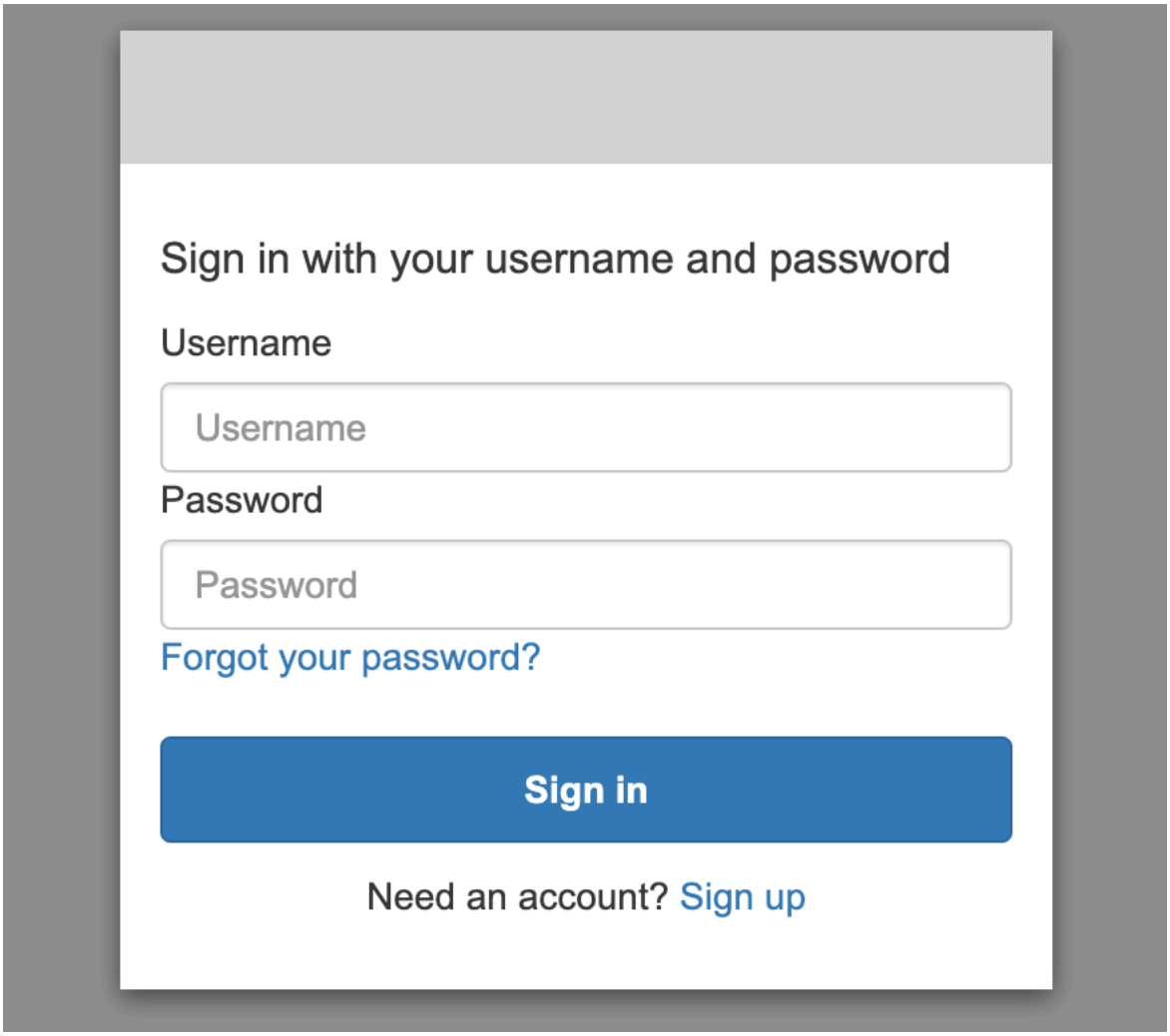
Método de autenticación	Antes de habilitar el control de acceso detallado	Después de habilitar el control de acceso detallado	Tareas administrativas
	pueden acceder al dominio.	de Internet pueden seguir accediendo al dominio.	2. Actualice sus clientes para proporcionar credenciales de autenticación básicas o credencia les de IAM, según la configuración de asignación de roles. 3. Desactive el período de migración. Los usuarios que envíen solicitudes sin autenticación básica o credencia les de IAM perderán el acceso al dominio.

Acceder a OpenSearch Dashboards como usuario maestro

El control de acceso detallado tiene un complemento de OpenSearch Dashboards que simplifica las tareas de administración. Puede utilizar Dashboards para administrar usuarios, roles, mapeos, grupos de acciones e inquilinos. Sin embargo, la página de inicio de sesión de OpenSearch Dashboards y el método de autenticación subyacente varían según cómo administre a los usuarios y cómo se haya configurado el dominio.

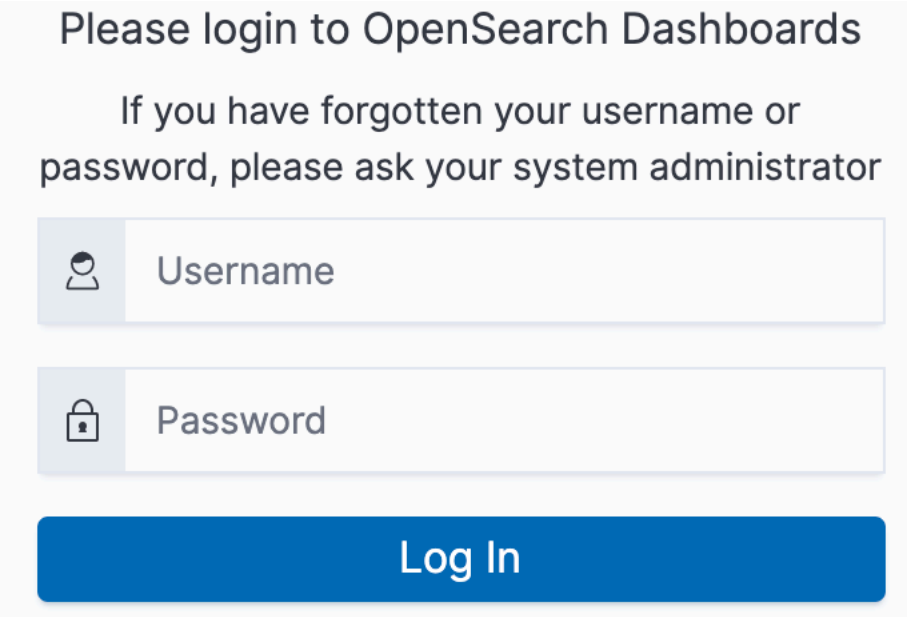
- Si desea utilizar IAM para la administración de usuarios, utilice the section called “Descripción de la autenticación de Amazon Cognito para OpenSearch Dashboards” para acceder a Dashboards. De lo contrario, Dashboards muestra una página de inicio de sesión no funcional. Consulte the section called “Limitaciones”.

  Con la autenticación de Amazon Cognito, uno de los roles asumidos del grupo de identidades debe coincidir con el rol de IAM especificado para el usuario maestro. Para obtener más información acerca de esta configuración, consulte the section called “(Opcional) Configuración de acceso pormenorizado” y the section called “Tutorial: Control de acceso detallado con la autenticación de Cognito”.



- Si decide utilizar la base de datos de usuarios interna, puede iniciar sesión en Dashboards con su nombre de usuario maestro y contraseña. Debe obtener acceso a Dashboards a través de HTTPS. La autenticación de Amazon Cognito y SAML para Dashboards reemplazan esta pantalla de inicio de sesión.

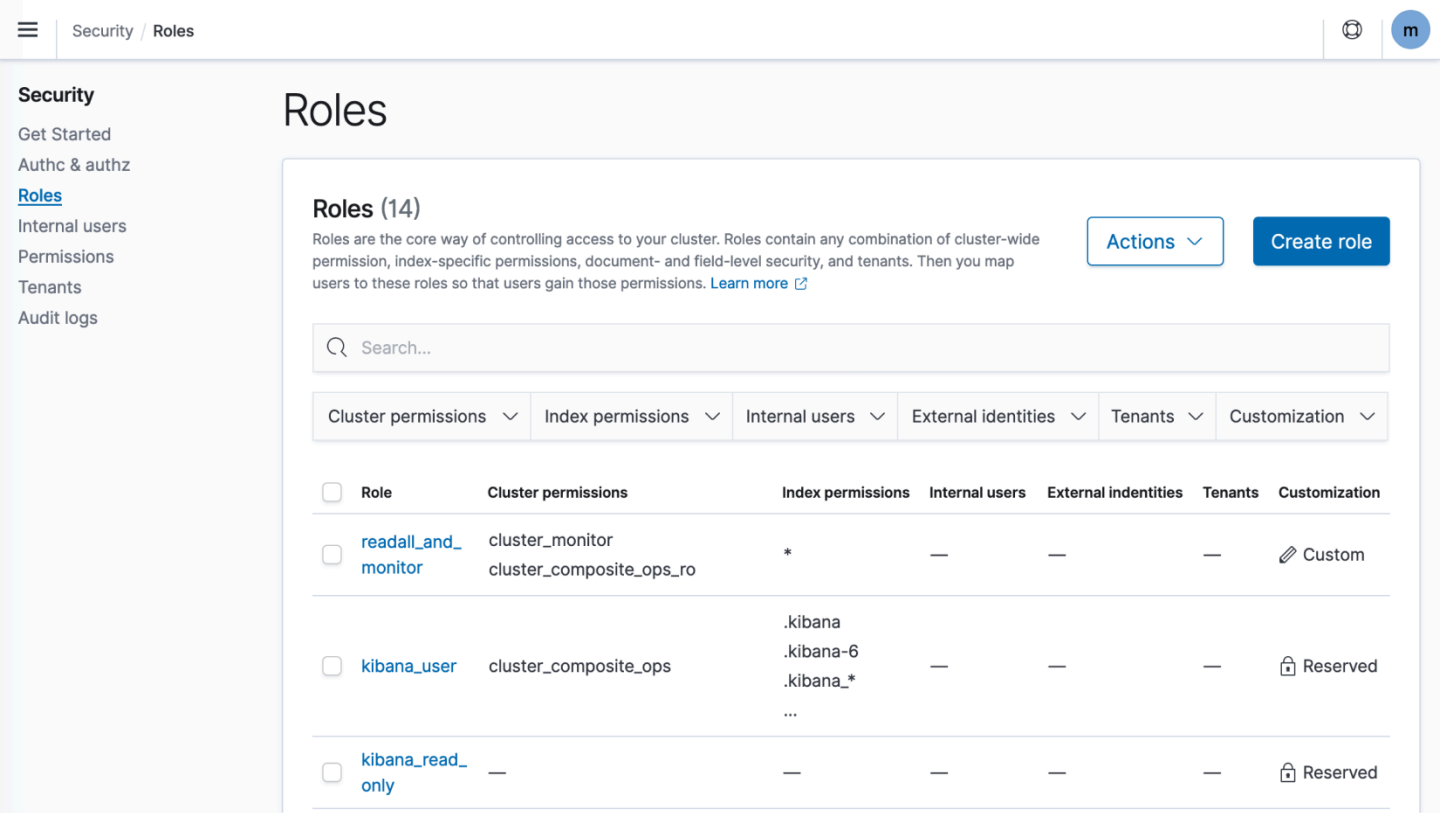
  Para obtener más información acerca de esta configuración, consulte the section called “Tutorial: Base de datos de usuarios interna con autenticación básica”.



- Si elige utilizar la autenticación SAML, puede iniciar sesión con credenciales de un proveedor de identidad externo. Para obtener más información, consulte the section called “Autenticación SAML para OpenSearch Dashboards”.

Administrar permisos

Tal y como se indica en the section called “Conceptos clave”, puede administrar los permisos de control de acceso detallado mediante roles, usuarios y mapeos. En esta sección, se describe cómo crear y aplicar esos recursos. Recomendamos que inicie sesión en Dashboards como usuario maestro para realizar estas operaciones.



> **Note**
>
> Los permisos que elige conceder a los usuarios varían ampliamente en función del caso de uso. No podemos cubrir de forma factible todos los escenarios de esta documentación. A medida que determina qué permisos conceder a los usuarios, asegúrese de hacer referencia al clúster de OpenSearch y a los permisos de índice mencionados en las siguientes secciones, y siga siempre el principio de privilegio mínimo.

Crear roles

Puede crear nuevos roles para el control de acceso detallado mediante OpenSearch Dashboards o la operación `_plugins/_security` en la API de REST. Para obtener más información, consulte Create roles (Crear roles).

El control de acceso detallado también incluye varios roles predefinidos. Clientes como OpenSearch Dashboards y Logstash realizan una amplia variedad de solicitudes a OpenSearch, lo que puede dificultar la creación manual de roles con el conjunto mínimo de permisos. Por ejemplo, el rol `opensearch_dashboards_user` incluye los permisos que un usuario necesita para trabajar con

patrones de índices, visualizaciones, paneles e inquilinos. Recomendamos mapearlo a cualquier rol de usuario o backend que tenga acceso a Dashboards, junto con los roles adicionales que permitan el acceso a otros índices.

Amazon OpenSearch Service no ofrece los siguientes roles de OpenSearch:

- `observability_full_access`
- `observability_read_access`
- `reports_read_access`
- `reports_full_access`

Amazon OpenSearch Service ofrece varios roles que no están disponibles en OpenSearch:

- `ultrawarm_manager`
- `ml_full_access`
- `cold_manager`
- `notifications_full_access`
- `notifications_read_access`

Seguridad en el nivel del clúster

Los permisos en el nivel del clúster incluyen la capacidad de ejecutar solicitudes generales como `_mget`, `_msearch` y `_bulk`, monitorear el estado o tomar instantáneas, etc. Administre estos permisos mediante la sección Permisos de clúster al crear un rol. Para obtener una lista completa de los permisos de nivel de clúster, consulte Permisos de cluster.

En lugar de permisos individuales, a menudo puede alcanzar la posición de seguridad deseada mediante una combinación de los grupos de acciones predeterminadas. Para obtener una lista de grupos de acciones en el nivel del clúster, consulte Nivel del clúster.

Seguridad en el nivel del índice

Los permisos en el nivel del índice incluyen la capacidad de crear nuevos índices, buscar índices, leer y escribir documentos, eliminar documentos, administrar alias y mucho más. Administre estos permisos con la sección Permisos de índice al crear un rol. Para obtener una lista completa de los permisos de nivel de índice, consulte Permisos de índice.

En lugar de permisos individuales, a menudo puede alcanzar la posición de seguridad deseada mediante una combinación de los grupos de acciones predeterminadas. Para obtener una lista de grupos de acciones en el nivel del índice, consulte Nivel del índice.

Seguridad en el nivel del documento

La seguridad en el nivel del documento permite restringir los documentos de un índice que puede ver un usuario. Al crear un rol, especifique un patrón de índice y una consulta de OpenSearch. Cualquier usuario que se mapee a ese rol únicamente podrá ver los documentos que coincidan con la consulta. La seguridad en el nivel del documento afecta al número de aciertos que se reciben al realizar una búsqueda.

Para obtener más información, consulte Seguridad en el nivel del documento.

Seguridad en el nivel del campo

La seguridad en el nivel del campo permite controlar qué campos de documento puede ver un usuario. Al crear un rol, agregue una lista de campos para incluirlos o excluirlos. Si incluye campos, los usuarios que mapee a ese rol solo podrán ver esos campos. Si excluye campos, podrán ver todos los campos excepto los excluidos. La seguridad en el nivel del campo afecta al número de campos que se incluyen en los aciertos al realizar una búsqueda.

Para obtener más información, consulte Seguridad en el nivel del campo.

Enmascarar campos

El enmascaramiento de campos es una alternativa a la seguridad en el nivel del campo que permite anonimizar los datos de un campo en lugar de eliminarlos por completo. Al crear un rol, agregue una lista de campos que se van a enmascarar. El enmascaramiento de campos afecta a si se podrá ver el contenido de un campo al realizar búsquedas.

> **Tip**
>
> Si aplica el enmascaramiento estándar a un campo, OpenSearch Service utiliza un hash aleatorio seguro que puede causar resultados de agregación inexactos. Para realizar agregaciones en campos enmascarados, utilice el enmascaramiento con base en patrones en su lugar.

Crear usuarios

Si ha habilitado la base de datos de usuarios interna, puede crear usuarios mediante OpenSearch Dashboards o mediante la operación `_plugins/_security` en la API REST. Para obtener más información, consulte Crear usuarios.

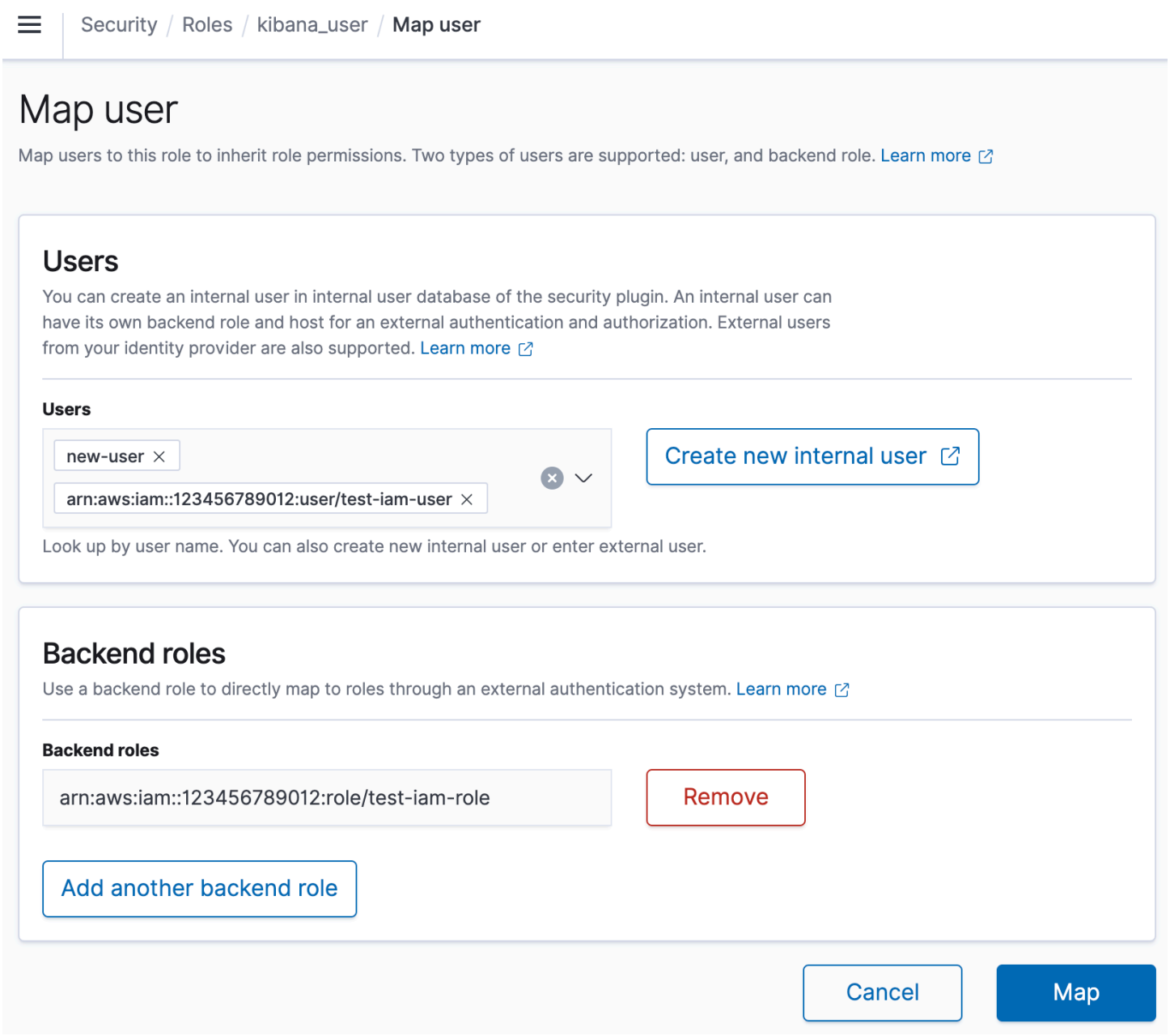
Si elige IAM para el usuario maestro, pase por alto esta parte de Dashboards. En su lugar, cree roles de IAM. Para obtener más información, consulte la Guía del usuario de IAM.

Mapear roles a usuarios

El mapeo de roles es el aspecto más crítico del control de acceso detallado. El control de acceso detallado tiene algunos roles predefinidos para ayudarlo a comenzar, pero a menos que mapee roles a los usuarios, cada solicitud al clúster dará lugar a un error de permisos.

Roles de backend pueden ayudar a simplificar el proceso de asignación de roles. En lugar de asignar el mismo rol a 100 usuarios individuales, puede asignar el rol a un único rol de backend que compartan los 100 usuarios. Los roles de backend pueden ser roles de IAM o cadenas arbitrarias.

- Especifique los usuarios, los ARN de usuario y las cadenas de usuario de Amazon Cognito en la sección Usuarios. Las cadenas de usuario de Cognito toman la forma de `Cognito/`*`user-pool-id`*`/`*`username`*.
- Especifique los roles de backend y los ARN de rol de IAM en la sección Roles de backend.



Puede mapear roles a usuarios mediante OpenSearch Dashboards o mediante la operación `_plugins/_security` de la API REST. Para obtener más información, consulte Mapear usuarios a roles.

Crear grupos de acciones

Los grupos de acciones son conjuntos de permisos que puede reutilizar en diferentes recursos. Puede crear nuevos grupos de acciones mediante OpenSearch Dashboards o la operación `_plugins/_security` en la API REST, aunque los grupos de acciones predeterminados son

suficientes para la mayoría de los casos de uso. Para obtener más información acerca de los grupos de acciones predeterminados, consulte Grupos de acciones predeterminados.

Varios inquilinos en OpenSearch Dashboards

Los inquilinos son espacios para guardar patrones de índices, visualizaciones, paneles y otros objetos de Dashboards. El uso de varios inquilinos en Dashboards permite compartir el trabajo de forma segura con otros usuarios de Dashboards (o mantenerlo en privado) y configurar inquilinos de forma dinámica. Puede controlar qué roles tienen acceso a un inquilino y si esos roles tienen acceso de lectura o de escritura. El inquilino global es el predeterminado. Para obtener más información, consulte Varios inquilinos en OpenSearch Dashboards multi-tenancy.

Para ver al inquilino actual o cambiar de inquilino

1. Acceda a OpenSearch Dashboards e inicie sesión.
2. Seleccione el icono de usuario en la parte superior derecha y elija Cambiar inquilinos.
3. Compruebe el inquilino antes de crear visualizaciones o paneles. Si desea compartir su trabajo con todos los demás usuarios de Dashboards, seleccione Global. Para compartir el trabajo con un subconjunto de usuarios de Dashboards, elija otro inquilino compartido. De lo contrario, seleccione Privado.

> **Note**
>
> OpenSearch Dashboards mantiene un índice independiente para cada arrendatario y crea una plantilla de índice llamada `tenant_template`. No elimine ni modifique el índice `tenant_template`, ya que podría provocar un mal funcionamiento de OpenSearch Dashboards si la asignación del índice de inquilinos está mal configurada.

Configuraciones recomendadas

Debido a la forma en que el control de acceso detallado interactúa con otras características de seguridad, recomendamos varias configuraciones de control de acceso detallado que funcionan bien en la mayoría de los casos de uso.

Descripción	Usuario maestro	Política de acceso al dominio

Utilice las credenciales de IAM para las llamadas a las API de OpenSearch y autenticación SAML para acceder a Dashboard s. Administre los roles de control de acceso detallado mediante Dashboards o la API de REST.	Rol o usuario de IAM	JSON
		{
		"Version":"2012-10-17",
		"Statement": [
		{
		"Effect": "Allow",
		"Principal": {
		"AWS": "*"
		},
		"Action": "es:ESHttp*",
		"Resource": "arn:aws:es: *us-east-1* :*111122223333* :domain/*domain-name* /*"
		}
		]
		}

Utilice credenciales de IAM o la autenticación básica para llamadas a las API de OpenSearch. Administre los roles de control de acceso detallado mediante Dashboards o la API REST.
 	
Esta configuración ofrece mucha flexibilidad, especialmente si tiene clientes de OpenSearch que solo admiten autenticación básica.
	Nombre de usuario y contraseña	JSON
		{
		"Version":"2012-10-17",
		"Statement": [
		{
		"Effect": "Allow",
		"Principal": {
		"AWS": "*"
		},
		"Action": "es:ESHttp*",
		"Resource": "arn:aws:es: *us-east-1* :*111122223333* :domain/*domain-name* /*"
		}
		]
		}

Descripción	Usuario maestro	Política de acceso al dominio
Si tiene un proveedor de identidad existente, utilice autenticación SAML para acceder a Dashboard s. En caso contrario, administre los usuarios de Dashboards en la base de datos de usuarios interna.		
Utilice las credenciales de IAM para las llamadas a las API de OpenSearc h y Amazon Cognito a fin de obtener acceso a Dashboards. Administr e los roles de control de acceso detallado mediante Dashboards o la API de REST.	Rol o usuario de IAM	JSON `{ "Version":"2012-10-17", "Statement": [ { "Effect": "Allow", "Principal": { "AWS": "*" }, "Action": "es:ESHttp*", "Resource": "arn:aws:es: us-east-1 :111122223333 :domain/domain-name /*" } ] }`

Descripción	Usuario maestro	Política de acceso al dominio

Utilice las credenciales de IAM para las llamadas a las API de OpenSearch y bloquee la mayor parte del acceso a Dashboard s. Administre los roles de control de acceso detallado mediante la API de REST.	Rol o usuario de IAM	JSON
		{
		"Version":"2012-10-17",
		"Statement": [
		{
		"Effect": "Allow",
		"Principal": {
		"AWS": "*"
		},
		"Action": "es:ESHttp*",
		"Resource": "arn:aws:es: *us-east-1* :*111122223333* :domain/*domain-name* /*"
		},
		{
		"Effect": "Deny",
		"Principal": {
		"AWS": "*"
		},
		"Action": "es:ESHttp*",
		"Resource": "arn:aws:es: *us-east-1* :*111122223333* :domain/*domain-name* /_dashboards*"
		}
		]
		}

Limitaciones

El control de acceso detallado tiene varias limitaciones importantes:

- El aspecto `hosts` de los mapeos de roles, que mapea roles a nombres de anfitrión o direcciones IP, no funciona si el dominio está dentro de una VPC. Aun así, puede mapear roles a usuarios y roles de backend.

- Si elige IAM para el usuario maestro y no habilita la autenticación de Amazon Cognito o SAML, Dashboards muestra una página de inicio de sesión no funcional.
- Si elige IAM para el usuario maestro, podrá crear usuarios en la base de datos de usuarios interna. Sin embargo, debido a que la autenticación básica de HTTP no está habilitada en esta configuración, todas las solicitudes firmadas con esas credenciales de usuario se rechazarán.
- Si utiliza SQL para consultar un índice al que no tiene acceso, recibirá un error debido a que no tiene los permisos necesarios. Si el índice no existe, recibirá un error debido a que ese índice no existe. Esta diferencia en los mensajes de error significa que puede confirmar si un índice existe o no, en el caso de que trate de adivinar su nombre.

  Para minimizar el problema, no incluya información confidencial en los nombres de índice. Para denegar por completo el acceso a SQL, agregue el siguiente elemento a la política de acceso al dominio:

```
{
  "Effect": "Deny",
  "Principal": {
    "AWS": [
      "*"
    ]
  },
  "Action": [
    "es:*"
  ],
  "Resource": "arn:aws:es:us-east-1:123456789012:domain/my-domain/_plugins/_sql"
}
```

- Si la versión de su dominio es 2.3 o superior y tiene activado el control de acceso detallado, configurar `max_clause_count` en 1 provocará problemas con su dominio. Le recomendamos configurar esta cuenta con un número superior.
- Si está habilitando un control de acceso detallado en un dominio en el que no está configurado un control de acceso detallado, en el caso de los orígenes de datos creados para consultas directas, tendrá que configurar las funciones de control de acceso detalladas. Para obtener más información sobre cómo configurar roles de acceso detallado, consulte Creación de integraciones de origen de datos de Amazon OpenSearch Service con Amazon S3.

Modificar el usuario maestro

Si olvida los detalles del usuario maestro, puede reconfigurarlo mediante la consola, la AWS CLI o la API de configuración.

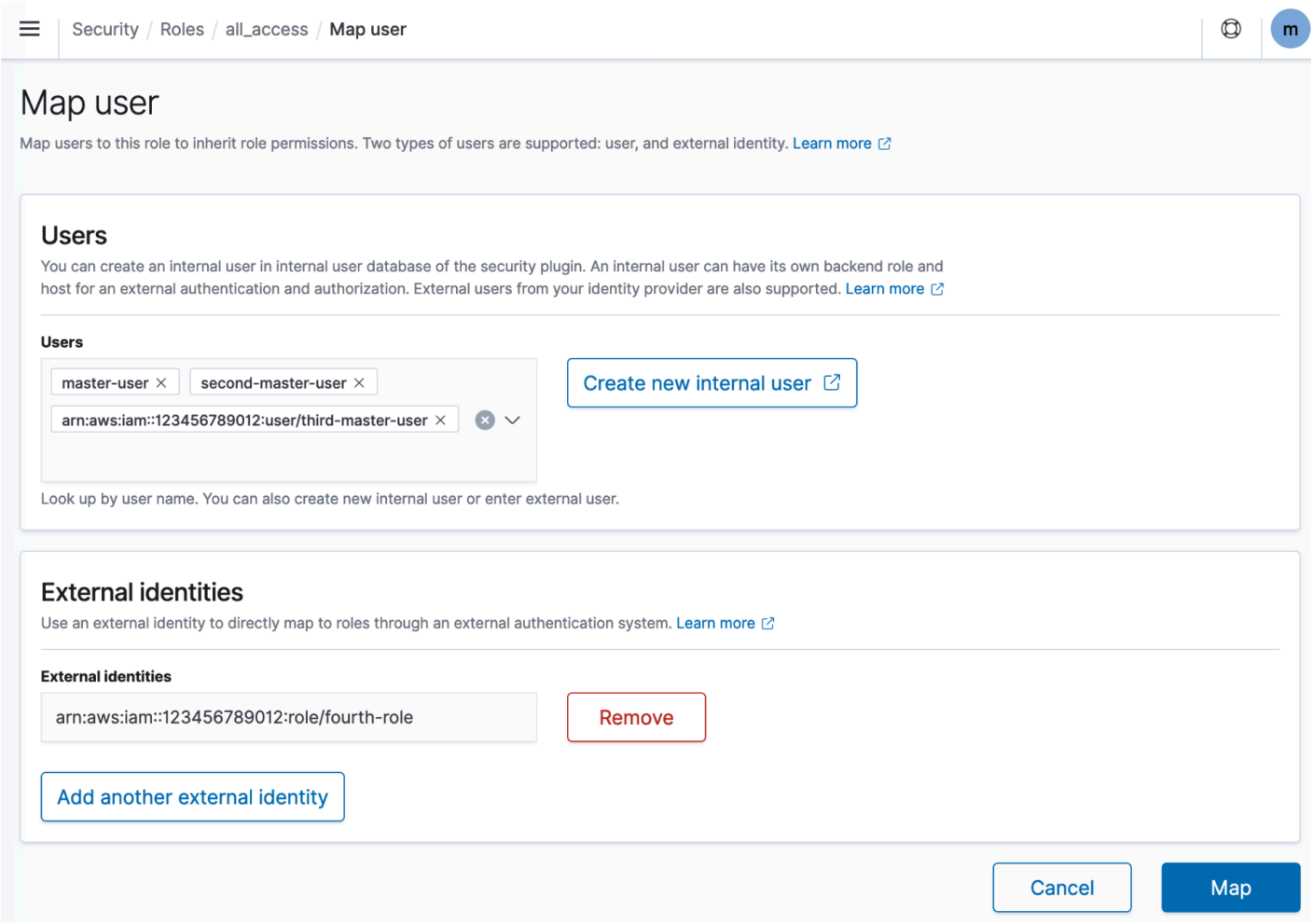
Para modificar el usuario maestro (consola)

1. Vaya a la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home/.
2. Seleccione el dominio y seleccione Acciones, Editar la configuración de seguridad.
3. Elika Establecer ARN de IAM como usuario maestro o bien Crear usuario maestro.
   - Si ha utilizado anteriormente un usuario maestro de IAM, el control de acceso detallado volverá a mapear el rol `all_access` al nuevo ARN de IAM que especifique.
   - Si ha utilizado previamente la base de datos de usuarios interna, el control de acceso detallado creará un usuario maestro. Puede utilizar el nuevo usuario maestro para eliminar el anterior.
   - Cambiar la base de datos de usuario interna a un usuario maestro de IAM no elimina ningún usuario de la base de datos de usuarios interna. En su lugar, simplemente desactiva la autenticación básica de HTTP. Elimine manualmente usuarios de la base de datos interna de usuarios o guárdelos en caso de que necesite volver a habilitar la autenticación básica de HTTP.
4. Seleccione Guardar cambios.

Usuarios maestros adicionales

Se designa un usuario maestro al crear un dominio. Sin embargo, si lo desea, puede utilizar este usuario maestro para crear usuarios maestros adicionales. Dispone de dos opciones: OpenSearch Dashboards o la API REST.

- En Dashboards, seleccione Seguridad, Roles y, a continuación, mapee el nuevo usuario maestro a los roles `all_access` y `security_manager`.



- Para utilizar la API de REST, envíe las siguientes solicitudes:

```
PUT _plugins/_security/api/rolesmapping/all_access
{
  "backend_roles": [
    "arn:aws:iam::123456789012:role/fourth-master-user"
  ],
  "hosts": [],
  "users": [
    "master-user",
    "second-master-user",
    "arn:aws:iam::123456789012:user/third-master-user"
  ]
}
```

```
PUT _plugins/_security/api/rolesmapping/security_manager
{
```

```
  "backend_roles": [
    "arn:aws:iam::123456789012:role/fourth-master-user"
  ],
  "hosts": [],
  "users": [
    "master-user",
    "second-master-user",
    "arn:aws:iam::123456789012:user/third-master-user"
  ]
}
```

Estas solicitudes reemplazan los mapeos de roles actuales. Por consiguiente, deber realizar las solicitudes `GET` primero para que pueda incluir todos los roles actuales en las solicitudes `PUT`. La API REST resulta especialmente útil si no puede obtener acceso a Dashboards y desea mapear un rol de IAM desde Amazon Cognito al rol `all_access`.

Instantáneas manuales

El control de acceso detallado presenta algunas complicaciones adicionales con la toma de instantáneas manuales. Para registrar un repositorio de instantáneas, aunque utilice la autenticación básica de HTTP para todos los demás fines, debe mapear el rol `manage_snapshots` a un rol de IAM que tenga permisos `iam:PassRole` para asumir `TheSnapshotRole`, tal como se define en the section called “Requisitos previos”.

A continuación, utilice ese rol de IAM para enviar una solicitud firmada al dominio, como se describe en the section called “Registrar un repositorio de instantáneas manuales”.

Integraciones

Si utiliza otros servicios de AWS con OpenSearch Service, debe proporcionar los roles de IAM para dichos servicios con los permisos adecuados. Por ejemplo, los flujos de entrega de Firehose suelen utilizar un rol de IAM denominado `firehose_delivery_role`. En Dashboards, cree un rol para el control de acceso detallado y mapee el rol de IAM a ese rol. En este caso, el nuevo rol necesita los siguientes permisos:

```
{
  "cluster_permissions": [
    "cluster_composite_ops",
    "cluster_monitor"
  ],
```

```
  "index_permissions": [{
    "index_patterns": [
      "firehose-index*"
    ],
    "allowed_actions": [
      "create_index",
      "manage",
      "crud"
    ]
  }]
}
```

Los permisos varían en función de las acciones que realice cada servicio. Una regla de AWS IoT o una función de AWS Lambda que indexa datos probablemente necesite permisos similares a Firehose, mientras que una función de Lambda que solo hace búsquedas puede utilizar un conjunto más limitado.

Diferencias en la API de REST

La API REST de control de acceso detallado presenta ligeras diferencias en función de la versión de OpenSearch/Elasticsearch. Antes de realizar una solicitud PUT, realice una solicitud GET para saber cómo es el cuerpo de la solicitud esperado. Por ejemplo, una solicitud GET para `_plugins/_security/api/user` devuelve todos los usuarios, que luego puede modificar y utilizar para realizar solicitudes PUT válidas.

En Elasticsearch 6.x, las solicitudes para crear usuarios tienen este aspecto:

```
PUT _opendistro/_security/api/user/new-user
{
  "password": "some-password",
  "roles": ["new-backend-role"]
}
```

En OpenSearch o Elasticsearch 7.x, las solicitudes tienen este aspecto (cambie `_plugins` a `_opendistro` si utiliza Elasticsearch):

```
PUT _plugins/_security/api/user/new-user
{
  "password": "some-password",
  "backend_roles": ["new-backend-role"]
}
```

Además, los inquilinos son propiedades de los roles en Elasticsearch 6.x:

```
GET _opendistro/_security/api/roles/all_access

{
  "all_access": {
    "cluster": ["UNLIMITED"],
    "tenants": {
      "admin_tenant": "RW"
    },
    "indices": {
      "*": {
        "*": ["UNLIMITED"]
      }
    },
    "readonly": "true"
  }
}
```

En OpenSearch y Elasticsearch 7.x, son objetos con su propio URI (cambie `_plugins` a `_opendistro` si utiliza Elasticsearch):

```
GET _plugins/_security/api/tenants

{
  "global_tenant": {
    "reserved": true,
    "hidden": false,
    "description": "Global tenant",
    "static": false
  }
}
```

Para obtener documentación acerca de la API de REST de OpenSearch, consulte la Referencia de la API de complementos de seguridad.

Tip

Si utiliza la base de datos de usuarios interna, puede utilizar curl para realizar solicitudes y probar el dominio. Pruebe los siguientes comandos de muestra:

```
curl -XGET -u 'master-user:master-user-password' 'domain-endpoint/_search'
```

```
curl -XGET -u 'master-user:master-user-password' 'domain-endpoint/_plugins/
_security/api/user'
```

Tutorial: Configurar un dominio con un usuario maestro de IAM y la autenticación de Amazon Cognito

Este tutorial cubre un caso de uso de habitual de Amazon OpenSearch Service de control de acceso detallado: un usuario maestro de IAM con autenticación de Amazon Cognito para OpenSearch Dashboards.

En el tutorial, configuraremos un rol de IAM maestro y un rol de IAM limitado, que luego asociaremos a los usuarios de Amazon Cognito. A continuación, el usuario maestro puede iniciar sesión en OpenSearch Dashboards, asignar el usuario limitado a un rol y utilizar el control de acceso detallado para limitar los permisos del usuario.



Aunque estos pasos utilizan el grupo de usuarios de Amazon Cognito para la autenticación, este mismo proceso básico funciona para cualquier proveedor de autenticación de Cognito que permita asignar diferentes roles de IAM a diferentes usuarios.

En este tutorial, deberá completar los siguientes pasos:

1. Crear roles de IAM maestros y limitados
2. Crear un dominio con la autenticación de Cognito
3. Configurar un grupo de usuarios y grupo de identidades de Cognito
4. Asignar roles en OpenSearch Dashboards
5. Evaluar los permisos

Paso 1: Crear roles de IAM maestros y limitados

Vaya a la consola de AWS Identity and Access Management (IAM) y cree dos roles independientes:

- `MasterUserRole`: es el usuario maestro, que tiene permisos completos para el clúster y, además, administra los roles y las asignaciones de los roles.
- `LimitedUserRole`: es un rol más restringido, al que concederá acceso limitado como usuario maestro.

Para obtener instrucciones acerca de cómo crear roles, consulte Creación de un rol mediante políticas de confianza personalizadas en la Guía del usuario de IAM.

Ambos roles deben tener la siguiente política de confianza, la cual permite que el grupo de identidades de Cognito asuma los roles:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Principal": {
      "Federated": "cognito-identity.amazonaws.com"
    },
    "Action": "sts:AssumeRoleWithWebIdentity",
    "Condition": {
      "StringEquals": {
        "cognito-identity.amazonaws.com:aud": "{identity-pool-id}"
      },
      "ForAnyValue:StringLike": {
        "cognito-identity.amazonaws.com:amr": "authenticated"
      }
    }
  }]
}
```

> **Note**
>
> Reemplace `identity-pool-id` por el identificador único de su grupo de identidades de Amazon Cognito. Por ejemplo, `us-east-1:0c6cdba7-3c3c-443b-a958-fb9feb207aa6`.

Paso 2: Crear un dominio con la autenticación de Cognito

Vaya a la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home/ y cree un dominio con la siguiente configuración:

- OpenSearch 1.0 o posterior, Elasticsearch 7.8 o posterior
- Acceso público
- Control de acceso detallado activado con `MasterUserRole` como el usuario maestro (creado en el paso anterior)
- Autenticación de Amazon Cognito para OpenSearch Dashboards habilitada. Para obtener instrucciones sobre cómo habilitar la autenticación de Cognito y seleccionar un grupo de usuarios e identidades, consulte the section called “Configuración de un dominio para utilizar la autenticación de Amazon Cognito”.
- La siguiente política de acceso al dominio:

  JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:root"
      },
      "Action": [
        "es:ESHttp*"
      ],
      "Resource": "arn:aws:es:us-east-1:111122223333:domain/{domain-name}/*"
    }
  ]
}
```

- Se requiere HTTPS para todo el tráfico al dominio
- Cifrado de nodo a nodo
- Cifrado de datos en reposo

Paso 3: configurar usuarios de Cognito

Mientras se crea su dominio, configure los usuarios maestros y limitados en Amazon Cognito siguiendo Crear un grupo de usuarios en la Guía para desarrolladores de Amazon Cognito. Por último, configure su grupo de identidades siguiendo los pasos de Crear un grupo de identidades en Amazon Cognito. El grupo de usuarios y el grupo de identidades deben estar en la misma Región de AWS.

Paso 4: Asignar roles en OpenSearch Dashboards

Ahora que sus usuarios están configurados, puede iniciar sesión en OpenSearch Dashboards como usuario maestro y asignar usuarios a roles.

1. Vuelva a la consola de OpenSearch Service y vaya a la URL de OpenSearch Dashboards para el dominio que creó. La URL tiene este formato: *domain-endpoint*`/_dashboards/`.
2. Inicie sesión con las credenciales de `master-user`.
3. Seleccione Agregar datos de muestra y agregue los datos de vuelos de muestra.
4. En el panel de navegación izquierdo, seleccione Seguridad, Roles, Crear rol.
5. Llame al rol `new-role`.
6. En Índice, especifique `opensearch_dashboards_sample_data_fli*` (`kibana_sample_data_fli*` en los dominios de Elasticsearch).
7. Para obtener Permisos de índice, seleccione leer.
8. Para Seguridad a nivel del documento, especifique la siguiente consulta:

```
{
  "match": {
    "FlightDelay": true
  }
}
```

9. Para la seguridad a nivel del campo, seleccione Excluir y especifique `FlightNum`.
10. Para Anonimización, especifique `Dest`.

11. Elija Crear.
12. Seleccione Usuarios asignados, Administrar mapeo. A continuación, agregue el Nombre de recurso de Amazon (ARN) para `LimitedUserRole` como una identidad externa y elija Asignar.
13. Vuelva a la lista de roles y elija opensearch_dashboards_user. Seleccione Usuarios asignados, Administrar mapeo. Agregue el ARN para `LimitedUserRole` como un rol de backend y elija Mapear.

Paso 5: Probar los permisos

Cuando los roles están asignados de manera correcta, puede iniciar sesión como el usuario limitado y probar los permisos.

1. En una nueva ventana privada del navegador, vaya a la URL de OpenSearch Dashboards del dominio, inicie sesión con las credenciales del `limited-user` y, a continuación, seleccione Explorar por mi cuenta.
2. Vaya a Herramientas para desarrolladores y ejecute la búsqueda predeterminada:

   ```
   GET _search
   {
     "query": {
       "match_all": {}
     }
   }
   ```

   Observe el error de permisos. `limited-user` no tiene permisos para ejecutar búsquedas en todo el clúster.
3. Ejecute otra búsqueda:

   ```
   GET opensearch_dashboards_sample_data_flights/_search
   {
     "query": {
       "match_all": {}
     }
   }
   ```

   Tenga en cuenta que todos los documentos coincidentes tienen un campo `FlightDelay` de `true`, un campo `Dest` anonimizado y ningún campo `FlightNum`.

4. En la ventana original del navegador, tras haber iniciado sesión como `master-user`, seleccione Herramientas para desarrolladores y, a continuación, realice las mismas búsquedas. Observe la diferencia en permisos, número de aciertos, documentos coincidentes y campos incluidos.

Tutorial: Configurar un dominio con la base de datos de usuarios interna y la autenticación básica de HTTP

Este tutorial explica otro caso de uso habitual de control de acceso detallado: un usuario maestro en la base de datos de usuarios interna y la autenticación básica de HTTP para OpenSearch Dashboards. A continuación, el usuario maestro puede iniciar sesión en OpenSearch Dashboards, crear un usuario interno, asignar el usuario a un rol y utilizar el control de acceso detallado para limitar los permisos del usuario.

En este tutorial, deberá completar los siguientes pasos:

1. Cree un dominio con un usuario maestro
2. Configure un usuario interno en OpenSearch Dashboards
3. Asignar roles en OpenSearch Dashboards
4. Evaluar los permisos

Paso 1: crear un dominio

Vaya a la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home/ y cree un dominio con la siguiente configuración:

- OpenSearch 1.0 o posterior, Elasticsearc 7.9 o posterior
- Acceso público
- Control de acceso detallado con un usuario maestro en la base de datos de usuarios interna (`TheMasterUser` en el resto de este tutorial)
- Autenticación de Amazon Cognito para Dashboards deshabilitada
- La siguiente política de acceso:

  JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
```

```
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:root"
      },
      "Action": [
        "es:ESHttp*"
      ],
      "Resource": "arn:aws:es:us-east-1:111122223333:domain/{domain-name}/*"
    }
  ]
}
```

- Se requiere HTTPS para todo el tráfico al dominio
- Cifrado de nodo a nodo
- Cifrado de datos en reposo

Paso 2: configurar un usuario interno en OpenSearch Dashboards

Ahora que tiene un dominio, puede iniciar sesión en OpenSearch Dashboards y crear un usuario interno.

1. Vuelva a la consola de OpenSearch Service y vaya a la URL de OpenSearch Dashboards para el dominio que creó. La URL tiene este formato: *domain-endpoint*/_dashboards/.
2. Inicie sesión con el `TheMasterUser`.
3. Seleccione Agregar datos de muestra y agregue los datos de vuelos de muestra.
4. En el panel de navegación izquierdo, seleccione Seguridad, Usuarios internos y Crear usuario interno.
5. Llame al usuario `new-user` y especifique una contraseña. A continuación, seleccione Crear.

Paso 3: asignar roles en OpenSearch Dashboards

Ahora que el usuario está configurado, puede asignarlo a un rol.

1. Permanezca en la sección Seguridad de OpenSearch Dashboards y elija Roles, Crear rol.
2. Llame al rol `new-role`.
3. Para Índice, especifique `opensearch_dashboards_sample_data_fli*` (`kibana_sample_data_fli*` en dominios Elasticsearch) para el patrón de índice.

4. Para el grupo de acciones, seleccione leer.
5. Para Seguridad a nivel del documento, especifique la siguiente consulta:

```
{
  "match": {
    "FlightDelay": true
  }
}
```

6. Para la seguridad a nivel del campo, seleccione Excluir y especifique `FlightNum`.
7. Para Anonimización, especifique `Dest`.
8. Elija Crear.
9. Seleccione Usuarios asignados, Administrar mapeo. A continuación, agregue `new-user` a Usuarios y elija Mapear.
10. Vuelva a la lista de roles y elija opensearch_dashboards_user. Seleccione Usuarios asignados, Administrar mapeo. A continuación, agregue `new-user` a Usuarios y elija Mapear.

Paso 4: probar los permisos

Cuando los roles están asignados de manera correcta, puede iniciar sesión como el usuario limitado y probar los permisos.

1. En una nueva ventana privada del navegador, vaya a la URL de OpenSearch Dashboards del dominio, inicie sesión con las credenciales del `new-user` y, a continuación, seleccione Explorar por mi cuenta.
2. Vaya a Herramientas para desarrolladores y ejecute la búsqueda predeterminada:

```
GET _search
{
  "query": {
    "match_all": {}
  }
}
```

   Observe el error de permisos. `new-user` no tiene permisos para ejecutar búsquedas en todo el clúster.
3. Ejecute otra búsqueda:

```
GET dashboards_sample_data_flights/_search
{
  "query": {
    "match_all": {}
  }
}
```

Tenga en cuenta que todos los documentos coincidentes tienen un campo `FlightDelay` de `true`, un campo `Dest` anonimizado y ningún campo `FlightNum`.

4. En la ventana original del navegador, inicie sesión como `TheMasterUser`, seleccione Herramientas para desarrolladores y realice las mismas búsquedas. Observe la diferencia en permisos, número de aciertos, documentos coincidentes y campos incluidos.

Validación de conformidad para Amazon OpenSearch Service

Auditores externos evalúan la seguridad y la conformidad de Amazon OpenSearch Service como parte de varios programas de conformidad de AWS. Entre estos se incluyen SOC, PCI e HIPAA.

Si tiene requisitos de cumplimiento, considere utilizar cualquier versión de OpenSearch o Elasticsearch 6.0 o posterior. Las versiones anteriores de Elasticsearch no ofrecen una combinación de cifrado de datos en reposo y cifrado de nodo a nodo y es poco probable que satisfagan sus necesidades. También podría considerar la posibilidad de utilizar cualquier versión de OpenSearch o Elasticsearch 6.7 o posterior si control de acceso detallado es importante para su caso de uso. En cualquier caso, elegir una versión específica de OpenSearch o Elasticsearch al crear un dominio no garantiza el cumplimiento.

Para saber si un Servicio de AWS está incluido en el ámbito de programas de conformidad específicos, consulte Servicios de AWS incluidos por programa de conformidad y elija el programa de conformidad que le interese. Para obtener información general, consulte Programas de conformidad de AWS.

Puedes descargar los informes de auditoría de terceros utilizando AWS Artifact. Para obtener más información, consulte Descarga de informes en AWS Artifact.

Su responsabilidad de conformidad al utilizar Servicios de AWS se determina en función de la confidencialidad de los datos, los objetivos de conformidad de su empresa, así como de la legislación y los reglamentos aplicables. Para obtener más información sobre la responsabilidad de conformidad al usar Servicios de AWS, consulte la Documentación de seguridad de AWS.

Resiliencia en Amazon OpenSearch Service

La infraestructura global de AWS se divide en Regiones de AWS y zonas de disponibilidad. Las Regiones de AWS proporcionan varias zonas de disponibilidad físicamente independientes y aisladas que se encuentran conectadas mediante redes con un alto nivel de rendimiento y redundancia, además de baja latencia. Con las zonas de disponibilidad, puede diseñar y utilizar aplicaciones y bases de datos que realizan una conmutación por error automática entre zonas de disponibilidad sin interrupciones. Las zonas de disponibilidad tienen una mayor disponibilidad, tolerancia a errores y escalabilidad que las infraestructuras tradicionales de centros de datos únicos o múltiples.

Para obtener más información sobre las Regiones de AWS y las zonas de disponibilidad, consulte la Infraestructura global de AWS.

Además de la infraestructura global de AWS, OpenSearch Service ofrece varias características que lo ayudan con sus necesidades de resiliencia y copia de seguridad de los datos.

- Dominios Multi-AZ y particiones replicadas
- Instantáneas automatizadas y manuales

Autenticación y autorización de JWT para Amazon OpenSearch Service

Amazon OpenSearch Service ahora le permite utilizar los JSON Web Tokens (JWT) para la autenticación y la autorización. Los JWT son tokens de acceso basados en JSON que se utilizan para conceder acceso de inicio de sesión único (SSO). Puede usar los JWT en OpenSearch Service para crear tokens de inicio de sesión único a fin de validar las solicitudes enviadas al dominio de OpenSearch Service. Para usar los JWT, debe tener habilitado un control de acceso detallado y debe proporcionar una clave pública válida con formato PEM de RSA o ECDSA. Para obtener más información sobre el control de acceso detallado, consulte Fine-grained access control in Amazon OpenSearch Service.

Puede configurar los JSON Web Tokens mediante la consola de OpenSearch Service, la AWS Command Line Interface (AWS CLI) o los AWS SDK.

Consideraciones

Antes de utilizar JWT con Amazon OpenSearch Service, debe tener en cuenta lo siguiente:

- Debido al tamaño de las claves públicas de RSA en formato PEM, se recomienda utilizar la consola de AWS para configurar la autenticación y la autorización de JWT.
- Debe proporcionar usuarios y roles válidos al especificar los campos de asuntos y roles de sus JWT; de lo contrario, se denegarán las solicitudes.
- OpenSearch 2.11 es la primera versión compatible que se puede utilizar para la autenticación de JWT.

Modificación de la política de acceso al dominio

Antes de poder configurar el dominio para usar la autenticación y la autorización de JWT, debe actualizar la política de acceso al dominio para permitir a los usuarios de JWT acceder al dominio. De lo contrario, se deniegan todas las solicitudes entrantes autorizadas de JWT. La política de acceso al dominio recomendada para proporcionar acceso total a los subrecursos (/*) es:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
      {
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
      },
      "Action": "es:ESHttp*",
      "Resource": "arn:aws:es:us-east-1:111122223333:domain/domain-name/*"
    }
  ]
}
```

Configuración de la autenticación y la autorización de JWT

Puede habilitar la autenticación y la autorización de JWT durante el proceso de creación del dominio o mediante la actualización de un dominio existente. Los pasos de configuración varían ligeramente según la opción que seleccione.

Los siguientes pasos explican cómo configurar un dominio existente para la autenticación y la autorización de JWT en la consola de OpenSearch Service:

1. En Configuración del dominio, vaya a Autenticación y autorización de JWT para OpenSearch y seleccione Activar la autenticación y la autorización JWT.
2. Configure la clave pública que se usará en el dominio. Para ello, puede cargar un archivo PEM que contenga una clave pública o ingresarla manualmente.

   > **Note**
   >
   > Si la clave cargada o ingresada no es válida, aparecerá una advertencia sobre el cuadro de texto que especifica el problema.

3. (Opcional) En Configuración adicional, puede configurar los siguientes campos opcionales:

   - Clave de asunto: puede dejar este campo vacío para usar la clave `sub` predeterminada para los JWT.
   - Clave de roles: puede dejar este campo vacío para usar la clave `roles` predeterminada para los JWT.

   Una vez hechos los cambios, guarde el dominio.

Uso de un JWT para enviar una solicitud de prueba

Tras crear un nuevo JWT con un par específico de sujeto y rol, puede enviar una solicitud de prueba. Para ello, utilice la clave privada para firmar la solicitud a través de la herramienta que creó el JWT. OpenSearch Service puede validar la solicitud entrante al verificar esta firma.

> **Note**
>
> Si especificó una clave de asunto o de roles personalizada para el JWT, debe utilizar los nombres de reclamaciones correctos para el JWT.

El siguiente es un ejemplo de cómo usar un token JWT para acceder a OpenSearch Service a través del punto de conexión de búsqueda del dominio:

```
curl -XGET "$search_endpoint" -H "Authorization: Bearer <JWT>"
```

Configuración de la autenticación y la autorización de JWT (AWS CLI)

El siguiente comando de la AWS CLI habilita la autenticación y la autorización de JWT para OpenSearch siempre que el dominio exista:

```
aws opensearch update-domain-config --domain-name <your_domain_name> --advanced-
security-options '{"JWTOptions":{"Enabled":true, "PublicKey": "<your_public_key>",
 "SubjectKey": "<your_subject_key>", "RolesKey": "<your_roles_key>"}}'
```

Configuración de la autenticación y la autorización de JWT (configuración a través de la API)

La siguiente solicitud a la API de configuración habilita la autenticación y la autorización de JWT para OpenSearch en un dominio existente:

```
POST https://es.us-east-1.amazonaws.com/2021-01-01/opensearch/domain/my-domain/config
{
  "AdvancedSecurityOptions": {
    "JWTOptions": {
      "Enabled": true,
      "PublicKey": "public-key",
      "RolesKey": "optional-roles-key",
      "SubjectKey": "optional-subject-key"
   }
  }
}
```

Generación de un par de claves

Para configurar JWT para el dominio de OpenSearch, necesitará proporcionar una clave pública en formato Privacy-Enhanced Mail (PEM). Amazon OpenSearch Service admite actualmente dos algoritmos de cifrado asimétrico cuando se utilizan los JWT: RSA y ECDSA.

Para crear un par de claves de RSA mediante la biblioteca openssl común, siga estos pasos:

1. `openssl genrsa -out privatekey.pem 2048`
2. `openssl rsa -in privatekey.pem -pubout -out publickey.pem`

En este ejemplo, el archivo `publickey.pem` contiene la clave pública para su uso con Amazon OpenSearch Service, mientras que `privatekey.pem` contiene la privada para firmar los JWT enviados al servicio. Además, tiene la opción de convertir la clave privada al formato `pkcs8` más utilizado si lo necesita para generar JWT.

Si utiliza el botón de carga para agregar un archivo PEM directamente a la consola, el archivo debe tener una extensión `.pem`; otras extensiones de archivo, como `.crt`, `.cert`, o `.key` no se admiten actualmente.

Seguridad de la infraestructura en Amazon OpenSearch Service

Como se trata de un servicio administrado, Amazon OpenSearch Service está protegido por la seguridad de red global de AWS. Para obtener información sobre los servicios de seguridad de AWS y sobre cómo AWS protege la infraestructura, consulte Seguridad en la nube de AWS. Para diseñar su entorno de AWS siguiendo las prácticas recomendadas de seguridad de infraestructura, consulte Protección de la infraestructura en Portal de seguridad de AWS Well-Architected Framework.

Puede utilizar llamadas a la API publicadas de AWS de OpenSearch Service a través de la red. Los clientes deben admitir lo siguiente:

- Seguridad de la capa de transporte (TLS). Exigimos TLS 1.2 y recomendamos TLS 1.3.
- Conjuntos de cifrado con confidencialidad directa total (PFS) como DHE (Ephemeral Diffie-Hellman) o ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). La mayoría de los sistemas modernos como Java 7 y posteriores son compatibles con estos modos.

Puede utilizar llamadas a la API publicadas de AWS para obtener acceso a la API de configuración de OpenSearch Service a través de la red. A fin de configurar la versión mínima requerida de TLS para aceptar, especifique el valor `TLSSecurityPolicy` en las opciones de punto de conexión del dominio:

```
aws opensearch update-domain-config --domain-name my-domain --domain-endpoint-options
 '{"TLSSecurityPolicy": "Policy-Min-TLS-1-2-2019-07"}'
```

Para obtener más información, consulte la Referencia de comandos de la AWS CLI.

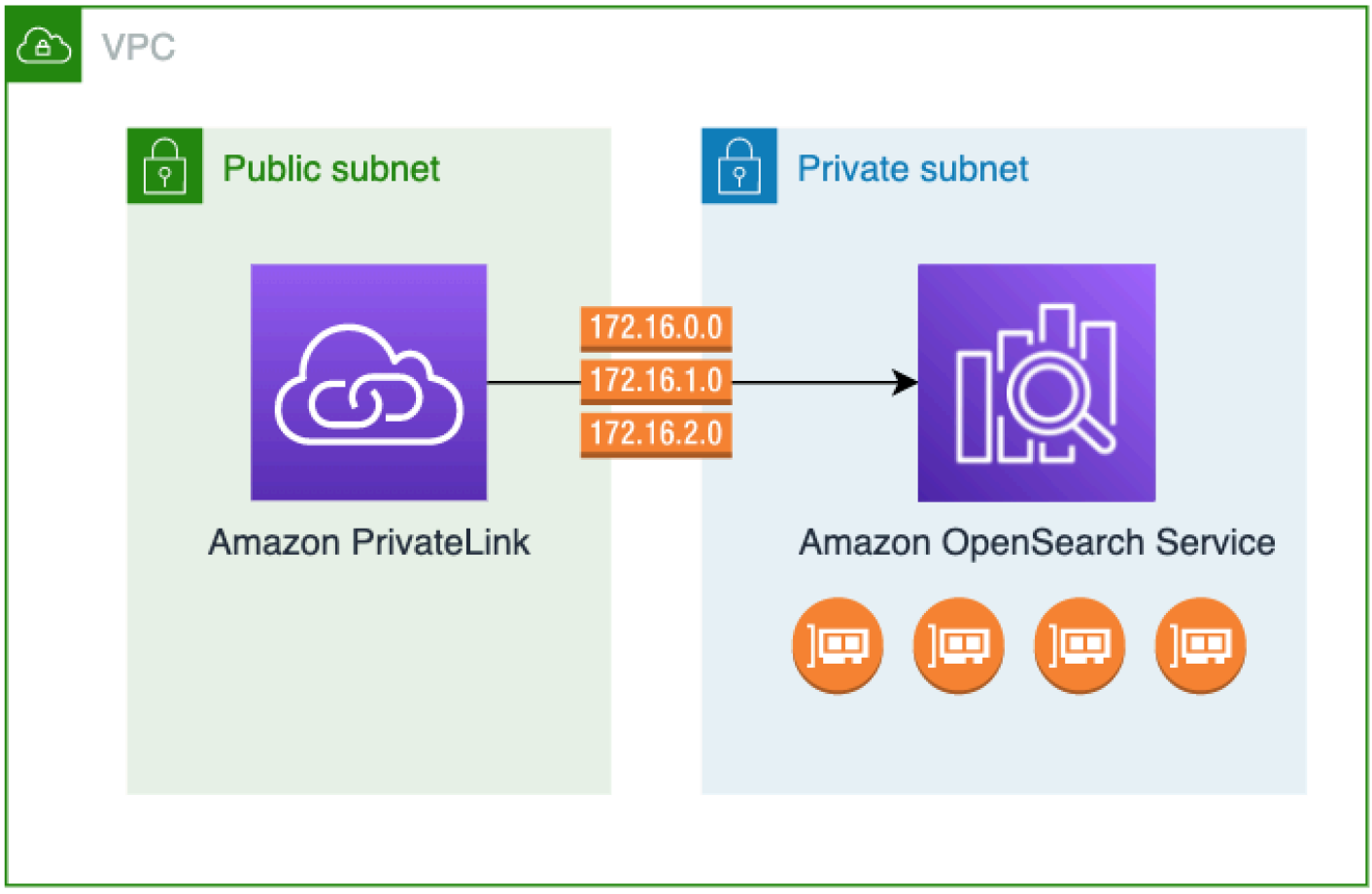
En función de la configuración de dominio, también podría ser necesario firmar solicitudes a las API de OpenSearch. Para obtener más información, consulte the section called “Realizar y firmar solicitudes de OpenSearch Service”.

OpenSearch Service admite dominios de acceso público, que pueden recibir solicitudes de cualquier dispositivo conectado a Internet y dominios de acceso mediante VPC, que están aislados de la Internet pública.

Acceda a Amazon OpenSearch Service mediante un punto de conexión de VPC administrado por OpenSearch Service (AWS PrivateLink)

Puede acceder a un dominio de Amazon OpenSearch Service mediante la configuración de un punto de conexión de VPC administrado por OpenSearch Service (con tecnología de AWS PrivateLink). Estos puntos de conexión de VPC crean una conexión privada entre la VPC y el Amazon OpenSearch Service. Puede acceder a los dominios de la VPC de Amazon OpenSearch Service como si estuvieran en su VPC, sin el uso de una puerta de enlace de Internet, un dispositivo NAT, una conexión VPN o una conexión Direct Connect. Las instancias de la VPC no necesitan direcciones IP públicas para acceder a Amazon OpenSearch Service.

Puede configurar los dominios de OpenSearch Service para exponer puntos de conexión adicionales que se ejecuten en subredes públicas o privadas dentro de la misma VPC, una VPC diferente o diferentes Cuentas de AWS. Esto le permite agregar una capa adicional de seguridad para acceder a sus dominios sin importar dónde se ejecuten, sin necesidad de administrar una infraestructura. El siguiente diagrama ilustra los puntos de conexión de VPC administrados por OpenSearch Service dentro de la misma VPC:



Esta conexión privada se establece mediante la creación de un punto de conexión de VPC de interfaz administrado por OpenSearch Service, con la tecnología de AWS PrivateLink. Creamos una interfaz de red de punto de conexión en cada subred habilitada para el punto de conexión de VPC de interfaz. Se trata de interfaces de red administradas por el servicio que sirven como punto de entrada para el tráfico destinado a OpenSearch Service. Se aplican precios de punto de conexión de la interfaz de AWS PrivateLink estándar para los puntos de conexión de VPC administrados por OpenSearch Service que se facturan en AWS PrivateLink.

Puede crear puntos de conexión de VPC para los dominios que ejecutan todas las versiones de OpenSearch y Elasticsearch heredado. Para obtener más información, consulte Acceso Servicios de AWS a través de AWS PrivateLink en la Guía de AWS PrivateLink.

Consideraciones y limitaciones del servicio OpenSearch

Antes de configurar un punto de conexión de VPC de interfaz para OpenSearch Service, consulte Acceso a un servicio de AWS mediante un punto de conexión de VPC de interfaz en la Guía de AWS PrivateLink.

Cuando utilice puntos de conexión de VPC administrados por OpenSearch Service, tenga en cuenta lo siguiente:

- Solo puede utilizar puntos de conexión de VPC de interfaz para conectarse a los dominios de la VPC. Los dominios públicos no son compatibles.
- Los puntos de conexión de VPC solo pueden conectarse a dominios dentro de la misma Región de AWS.
- HTTPS es el único protocolo compatible para los puntos de conexión de VPC. No se permite el protocolo HTTP.
- OpenSearch Service permite realizar llamadas a todas las operaciones de la API de OpenSearch compatibles a través de un punto de conexión de VPC de interfaz.
- Puede configurar un máximo de 50 puntos de conexión por cuenta y un máximo de 10 puntos de conexión por dominio. Un solo dominio puede tener un máximo de 10 entidades principales autorizadas.
- En la actualidad, no puede utilizar AWS CloudFormation para crear puntos de conexión de VPC de interfaz.
- Solo puede crear puntos de conexión de VPC de interfaz a través de la consola de OpenSearch Service o mediante la API de OpenSearch Service. No puede crear un punto de conexión de VPC para OpenSearch Service mediante la consola de Amazon VPC.

- No se puede acceder a los puntos de conexión de VPC administrados por OpenSearch Service desde Internet. Solo se puede acceder a un punto de conexión de VPC administrado por OpenSearch Service dentro de la VPC donde se aprovisiona el punto de conexión o en cualquier VPC interconectada con la VPC donde se aprovisiona el punto de conexión, según lo permitan las tablas de enrutamiento y los grupos de seguridad.
- Las políticas de punto de conexión de VPC no son compatibles con OpenSearch Service. Puede asociar un grupo de seguridad a las interfaces de red de los puntos de conexión para controlar el tráfico a OpenSearch Service a través del punto de conexión de VPC de interfaz.
- Su rol vinculado al servicio debe estar en la misma cuenta AWS que utilizó para crear el punto de conexión de VPC.
- Para crear, actualizar y eliminar el punto de conexión de VPC de OpenSearch Service, debe tener los siguientes permisos de Amazon EC2 además de los permisos de Amazon OpenSearch Service:
  - `ec2:CreateVpcEndpoint`
  - `ec2:DescribeVpcEndpoints`
  - `ec2:ModifyVpcEndpoint`
  - `ec2:DeleteVpcEndpoints`
  - `ec2:CreateTags`
  - `ec2:DescribeTags`
  - `ec2:DescribeSubnets`
  - `ec2:DescribeSecurityGroups`
  - `ec2:DescribeVpcs`

> **Note**
>
> Actualmente, no se puede limitar la creación de puntos de conexión de VPC a OpenSearch Service. Estamos trabajando para que esto sea posible en una futura actualización.

Proporcionar el acceso a un dominio

Si la VPC a la que desea que su dominio acceda se encuentra en otra Cuenta de AWS, debe autorizarla desde la cuenta del propietario antes de crear un punto de conexión de VPC de interfaz.

Para permitir que una VPC de otra Cuenta de AWS acceda a su dominio

1. Abra la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home/.
2. En el panel de navegación, seleccione Dominios y abra el dominio al que desea proporcionar acceso.
3. Vaya a la pestaña Puntos de conexión de VPC, que muestra las cuentas y las VPC correspondientes que tienen acceso a su dominio.
4. Seleccione Autorizar entidad principal.
5. Escriba el ID de Cuenta de AWS de la cuenta que accederá al dominio. Este paso autoriza a la cuenta especificada a crear puntos de conexión de VPC en el dominio.
6. Seleccione Autorizar.

Creación de un punto de conexión de VPC de interfaz para un dominio de VPC

Puede crear un punto de conexión de VPC de interfaz para OpenSearch Service mediante la consola de OpenSearch Service o la AWS Command Line Interface (AWS CLI).

Para crear un punto de conexión de VPC de interfaz para un dominio de OpenSearch Service

1. Abra la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home/.
2. En el panel de navegación izquierdo, seleccione Puntos de conexión de VPC.
3. Seleccione Crear punto de conexión.
4. Seleccione si desea conectar un dominio en la Cuenta de AWS actual o en otra Cuenta de AWS.
5. Seleccione el dominio al que se conecta con este punto de conexión. Si el dominio está en la Cuenta de AWS actual, use el menú desplegable para elegir el dominio. Si el dominio se encuentra en otra cuenta, escriba el nombre de recurso de Amazon (ARN) del dominio para conectarse. Para elegir un dominio en una cuenta diferente, el propietario debe otorgarle acceso al dominio.
6. En VPC, seleccione la VPC desde la que accederá a OpenSearch Service.
7. En Subredes, seleccione una o más subredes desde las que accederá a OpenSearch Service.
8. En Grupos de seguridad, seleccione los grupos de seguridad que deban asociarse a las interfaces de red del punto de conexión. Este es un paso fundamental para limitar los puertos, los protocolos y los orígenes del tráfico entrante que autoriza para el punto de conexión. Las

reglas del grupo de seguridad deben permitir a los recursos que utilizarán el punto de conexión de VPC comunicarse con OpenSearch Service para comunicarse con la interfaz de red del punto de conexión.

9. Seleccione Crear punto de conexión. El punto de conexión debería estar activo en un plazo de 2 a 5 minutos.

Uso de puntos de conexión de VPC administrados por OpenSearch Service mediante la API de configuración

Utilice las siguientes operaciones de la API para crear y administrar puntos de conexión de VPC administrados por OpenSearch Service.

- CreateVpcEndpoint
- ListVpcEndpoints
- UpdateVpcEndpoint
- DeleteVpcEndpoint

Utilice las siguientes operaciones de la API para administrar el acceso de los puntos de conexión a los dominios de la VPC:

- AuthorizeVpcEndpointAccess
- ListVpcEndpointAccess
- ListVpcEndpointsForDomain
- RevokeVpcEndpointAccess

Autenticación SAML para OpenSearch Dashboards

La autenticación SAML para OpenSearch Dashboards permite utilizar su proveedor de identidades existente con el fin de ofrecer inicio de sesión único (SSO) para Dashboards en dominios de Amazon OpenSearch Service que ejecutan OpenSearch o Elasticsearch 6.7 o versiones posteriores. Para utilizar la autenticación SAML, debe habilitar el control de acceso detallado.

En lugar de autenticarse a través de Amazon Cognito o la base de datos de usuario interna, la autenticación SAML para OpenSearch Dashboards permite utilizar proveedores de identidad de terceros a fin de iniciar sesión en Dashboards, administrar un control de acceso detallado, buscar

sus datos y crear visualizaciones. OpenSearch Service admite proveedores que utilizan el estándar SAML 2.0, como Okta, Keycloak, Active Directory Federation Services (ADFS), Auth0 y AWS IAM Identity Center.

La autenticación SAML para Dashboards es solo para acceder a OpenSearch Dashboards a través de un navegador web. Las credenciales de SAML no permiten realizar solicitudes HTTP directas a las API de OpenSearch o Dashboards.

Información general de la configuración de SAML

En esta documentación se supone que tiene un proveedor de identidad existente y que está familiarizado con él en cierta medida. No podemos proporcionar pasos de configuración detallados para su proveedor exacto, solo para su dominio de OpenSearch Service.

El flujo de inicio de sesión de OpenSearch Dashboards puede adoptar una de dos formas:

- Proveedor de servicios (SP) iniciado: navegue al panel (por ejemplo, `https://my-domain.us-east-1.es.amazonaws.com/_dashboards`), que lo redirige a la pantalla de inicio de sesión. Después de iniciar sesión, el proveedor de identidades lo redirige al panel.
- Proveedor de identidades (IdP) iniciado: navegue hasta su proveedor de identidades, inicie sesión y elija OpenSearch Dashboards en un directorio de aplicaciones.

OpenSearch Service proporciona dos direcciones URL de inicio de sesión único, iniciadas por SP e iniciadas por IdP, pero solo necesita la que coincida con el flujo de inicio de sesión de OpenSearch Dashboards.

Independientemente del tipo de autenticación que utilice, el objetivo es iniciar sesión a través de su proveedor de identidades y recibir una aserción SAML que contenga su nombre de usuario (requerido) y cualquier rol backend (opcional, pero recomendado). Esta información permite el control de acceso detallado para asignar permisos a usuarios de SAML. En los proveedores de identidad externos, los roles backend se denominan generalmente “roles” o “grupos”.

Consideraciones

Tenga en cuenta lo siguiente cuando configure la autenticación SAML:

- Debido al tamaño del archivo de metadatos del IdP, se recomienda utilizar la consola de AWS para configurar la autenticación SAML.

- Los dominios solo admiten un método de autenticación del panel a la vez. Si tiene habilitada la autenticación de Amazon Cognito para OpenSearch Dashboards, debe deshabilitarla para poder habilitar la autenticación SAML.
- Si utiliza un equilibrador de carga de red con SAML, primero debe crear un punto de conexión personalizado. Para obtener más información, consulte ???.
- Las políticas de control de servicios (SCP) no se aplicarán ni evaluarán en el caso de las identidades que no sean de IAM (como SAML en Amazon OpenSearch sin servidor y SAML y la autorización de usuario interna básica para Amazon OpenSearch Service).

Autenticación SAML para dominios de VPC

SAML no requiere comunicación directa entre el proveedor de identidades y el proveedor de servicios. Por lo tanto, aunque el dominio de OpenSearch esté alojado en una VPC privada, se puede seguir utilizando SAML, siempre que el navegador pueda comunicarse tanto con el clúster de OpenSearch como con el proveedor de identidades. En esencia, el navegador actúa como intermediario entre el proveedor de identidades y el proveedor de servicios. Para ver un útil diagrama que explica el flujo de autenticación SAML, consulte la documentación de Okta.

Modificación de la política de acceso al dominio

Antes de configurar la autenticación SAML, debe actualizar la política de acceso al dominio para permitir a los usuarios de SAML acceder al dominio. De lo contrario, aparecerán errores de acceso denegado.

Recomendamos la siguiente política de acceso al dominio, que proporciona acceso completo a los subrecursos (`/*`) en el dominio:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
      },
      "Action": "es:ESHttp*",
```

```
      "Resource": "arn:aws:es:us-east-1:111122223333:domain/domain-name/*"
    }
  ]
}
```

Para hacer que la política sea más restrictiva, puede agregar una condición de dirección IP a la política. Esta condición limita el acceso únicamente a la subred o rango de direcciones IP especificados. Por ejemplo, la siguiente política solo permite el acceso desde la subred 192.0.2.0/24:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
      },
      "Action": [
        "es:ESHttp*"
      ],
      "Condition": {
        "IpAddress": {
          "aws:SourceIp": [
            "192.0.2.0/24"
          ]
        }
      },
      "Resource": "arn:aws:es:us-east-1:111122223333:domain/domain-name/*"
    }
  ]
}
```

> **Note**
>
> Una política de acceso a un dominio abierto requiere que se habilite un control de acceso detallado en el dominio; de lo contrario, aparece el siguiente error:

`To protect domains with public access, a restrictive policy or fine-grained access control is required.`
Si tiene un usuario maestro o un usuario interno configurado con una contraseña segura, mantener la política abierta mientras usa un control de acceso detallado podría ser aceptable desde el punto de vista de la seguridad. Para obtener más información, consulte ???.

Configuración de la autenticación iniciada por proveedor de servicios o por proveedor de identidades

Estos pasos explican cómo habilitar la autenticación SAML con autenticación iniciada por SP o IdP para OpenSearch Dashboards. Para conocer el paso adicional necesario para habilitar ambas, consulte Configuración de la autenticación iniciada por proveedor de servicios o por proveedor de identidades.

Paso 1: habilitar la autenticación SAML

Puede habilitar la autenticación SAML bien durante la creación del dominio o bien seleccionando Acciones, Editar la configuración de seguridad en un dominio existente. Los siguientes pasos varían ligeramente según lo que seleccione.

En la configuración del dominio, en Autenticación SAML para OpenSearch Dashboards/Kibana, seleccione Habilitar autenticación SAML.

Paso 2: configurar el proveedor de identidades

Ejecute los siguientes pasos en función del momento de configuración de la autenticación SAML.

Si va a crear un nuevo dominio

Si se encuentra en proceso de crear un nuevo dominio, OpenSearch Service aún no puede generar un ID de entidad del proveedor de servicios ni URL de SSO. El proveedor de identidades requiere estos valores para habilitar correctamente la autenticación SAML, pero solo se pueden generar después de crear el dominio. Para evitar esta interdependencia durante la creación del dominio, puede proporcionar valores temporales en la configuración del IdP con el fin de generar los metadatos necesarios, y luego actualizarlos una vez que el dominio esté activo.

Si utiliza un punto de conexión personalizado, puede inferir cuáles serán las URL. Por ejemplo, si el punto de conexión personalizado es `www.custom-endpoint.com`, el ID de entidad del proveedor

de servicios seráwww.*custom-endpoint*.com, la URL de SSO iniciado por IdP será www.*custom-endpoint*.com/_dashboards/_opendistro/_security/saml/acs/idpinitiated, y la URL de SSO iniciado por SP será www.*custom-endpoint*.com/_dashboards/_opendistro/_security/saml/acs. Puede utilizar los valores para configurar el proveedor de identidades antes de crear el dominio. Consulte la siguiente sección para ver ejemplos.

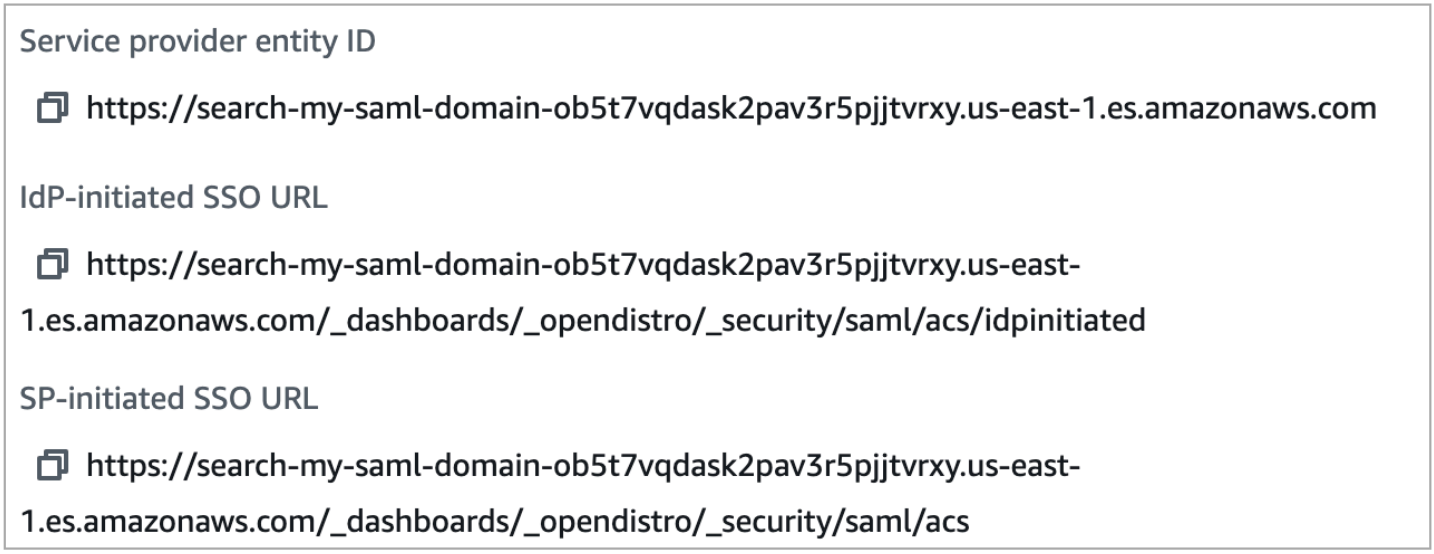
> **Note**
>
> No puede iniciar sesión con un punto de conexión de doble pila porque el FQDN de una solicitud HTTP es diferente del FQDN de una solicitud SAML. Un administrador de OpenSearch tendrá que configurar un punto de conexión personalizado y establecer el valor de CNAME como punto de conexión de doble pila si quiere iniciar sesión con un punto de conexión de doble pila.

Si no utiliza un punto de conexión personalizado, puede introducir valores temporales en el IdP para generar los metadatos necesarios, y luego actualizarlos una vez que el dominio esté activo.

Por ejemplo, en Okta puede introducir https://*temp-endpoint*.amazonaws.com en los campos URL de inicio de sesión único y URI de audiencia (ID de entidad del SP), lo que permite generar los metadatos. Luego, una vez que el dominio esté activo, puede recuperar los valores correctos desde OpenSearch Service y actualizarlos en Okta. Para obtener instrucciones, consulte the section called “Paso 6: Actualizar las URL del IdP”.

Si va a editar un dominio existente

Si va habilitar la autenticación SAML en un dominio existente, copie el ID de entidad del proveedor de servicios y una de las URL de SSO. Para obtener orientación sobre qué URL debe utilizar, consulte the section called “Información general de la configuración de SAML”.



Utilice los valores para configurar el proveedor de identidades. Esta es la parte más compleja del proceso, y, desafortunadamente, la terminología y los pasos varían enormemente según el proveedor. Consulte la documentación de su proveedor.

En Okta, por ejemplo, crea una aplicación web SAML 2.0. En URL de inicio de sesión único, especifique la URL de SSO. Para URI de audiencia (ID de identidad del SP), especifique el ID de entidad del SP.

En lugar de usuarios y roles de backend, Okta tiene usuarios y grupos. En Instrucciones de atributo de grupo, se recomienda agregar `role` al campo Nombre y la expresión regular `.+` al campo Filtro. Esta instrucción indica al proveedor de identidades de Okta que incluya todos los grupos de usuarios bajo el campo `role` de la aserción SAML después de que un usuario se autentica.

En el Centro de identidades de IAM, especifique el ID de la entidad del SP como la audiencia de SAML de la aplicación. También debe especificar las siguientes asignaciones de atributos: `Subject=${user:subject}:format=unspecified` y `Role=${user:groups}:format=uri`.

En Auth0, crea una aplicación web regular y habilita el complemento SAML 2.0. En Keycloak, crea un cliente.

Paso 3: Importar metadatos del IdP

Después de configurar el proveedor de identidades, genera un archivo de metadatos de IdP. Este archivo XML contiene información sobre el proveedor, como un certificado TLS, puntos de conexión de inicio de sesión único y el ID de entidad del proveedor de identidad.

Copie el contenido del archivo de metadatos del IdP y péguelo en el campo Metadatos del IdP en la consola de OpenSearch Service. Alternativamente, elija Import from XML file (Importar desde archivo XML) y cargue el archivo. El archivo de metadatos debe tener un aspecto similar al siguiente:

```
<?xml version="1.0" encoding="UTF-8"?>
<md:EntityDescriptor entityID="entity-id"
 xmlns:md="urn:oasis:names:tc:SAML:2.0:metadata">
  <md:IDPSSODescriptor WantAuthnRequestsSigned="false"
 protocolSupportEnumeration="urn:oasis:names:tc:SAML:2.0:protocol">
    <md:KeyDescriptor use="signing">
      <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
        <ds:X509Data>
          <ds:X509Certificate>tls-certificate</ds:X509Certificate>
        </ds:X509Data>
      </ds:KeyInfo>
    </md:KeyDescriptor>
    <md:NameIDFormat>urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified</
md:NameIDFormat>
    <md:NameIDFormat>urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress</
md:NameIDFormat>
    <md:SingleSignOnService Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
 Location="idp-sso-url"/>
    <md:SingleSignOnService Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-
Redirect" Location="idp-sso-url"/>
  </md:IDPSSODescriptor>
</md:EntityDescriptor>
```

Paso 4: configurar los campos SAML

Después de introducir los metadatos del IdP, configure los siguientes campos adicionales en la consola de OpenSearch Service:

- ID de entidad del IdP: copie el valor de la propiedad `entityID` del archivo de metadatos y péguelo en este campo. Muchos proveedores de identidades también muestran este valor como parte de un resumen posterior a la configuración. Algunos proveedores lo llaman el “emisor”.
- Nombre de usuario maestro de SAML y Rol de backend maestro de SAML: el usuario o rol de backend que se especifique recibe permisos completos para el clúster, equivalentes a un nuevo usuario maestro, pero solo se pueden utilizar esos permisos en OpenSearch Dashboards.

  Por ejemplo, en Okta, es posible que tenga un usuario `jdoe` que pertenece al grupo `admins`. Si agrega `jdoe` al nombre de usuario maestro de SAML, solo ese usuario recibe permisos

completos. Si agrega `admins` Rol de backend maestro de SAML, cualquier usuario que pertenezca al grupo `admins` recibe permisos completos.

> **Note**
>
> El contenido de la aserción SAML debe coincidir exactamente con las cadenas que se utilicen para el nombre de usuario maestro de SAML y el rol maestro de SAML. Algunos proveedores de identidad agregan un prefijo antes de sus nombres de usuario, lo que puede provocar una coincidencia difícil de diagnosticar. En la interfaz de usuario del proveedor de identidades, es posible que vea `jdoe`, pero la aserción SAML podría contener `auth0|jdoe`. Utilice siempre la cadena de la aserción SAML.

Muchos proveedores de identidades permiten ver una aserción de muestra durante el proceso de configuración, y herramientas como Rastreo SAML pueden ayudarlo a examinar y solucionar problemas del contenido de las aserciones reales. Las aserciones se ven algo similar a lo siguiente:

```
<?xml version="1.0" encoding="UTF-8"?>
<saml2:Assertion ID="id67229299299259351343340162"
 IssueInstant="2020-09-22T22:03:08.633Z" Version="2.0"
  xmlns:saml2="urn:oasis:names:tc:SAML:2.0:assertion">
  <saml2:Issuer Format="urn:oasis:names:tc:SAML:2.0:nameid-format:entity">idp-issuer</
saml2:Issuer>
  <saml2:Subject>
    <saml2:NameID Format="urn:oasis:names:tc:SAML:1.1:nameid-
format:unspecified">username</saml2:NameID>
    <saml2:SubjectConfirmation Method="urn:oasis:names:tc:SAML:2.0:cm:bearer">
      <saml2:SubjectConfirmationData NotOnOrAfter="2020-09-22T22:08:08.816Z"
 Recipient="domain-endpoint/_dashboards/_opendistro/_security/saml/acs"/>
    </saml2:SubjectConfirmation>
  </saml2:Subject>
  <saml2:Conditions NotBefore="2020-09-22T21:58:08.816Z"
 NotOnOrAfter="2020-09-22T22:08:08.816Z">
    <saml2:AudienceRestriction>
      <saml2:Audience>domain-endpoint</saml2:Audience>
    </saml2:AudienceRestriction>
  </saml2:Conditions>
  <saml2:AuthnStatement AuthnInstant="2020-09-22T19:54:37.274Z">
    <saml2:AuthnContext>
```

```
 <saml2:AuthnContextClassRef>urn:oasis:names:tc:SAML:2.0:ac:classes:PasswordProtectedTransport<
saml2:AuthnContextClassRef>
    </saml2:AuthnContext>
  </saml2:AuthnStatement>
  <saml2:AttributeStatement>
    <saml2:Attribute Name="role" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
format:unspecified">
      <saml2:AttributeValue
        xmlns:xs="http://www.w3.org/2001/XMLSchema"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xsi:type="xs:string">GroupName Match Matches regex ".+" (case-sensitive)
      </saml2:AttributeValue>
    </saml2:Attribute>
  </saml2:AttributeStatement>
</saml2:Assertion>
```

Paso 5: (opcional) configurar ajustes adicionales

En Configuración adicional, configure los siguientes campos opcionales:

- Clave de asunto: puede dejar este campo vacío con el fin de utilizar el elemento `NameID` de la aserción SAML para el nombre de usuario. Si su aserción no utiliza este elemento estándar y, en su lugar, incluye el nombre de usuario como un atributo personalizado, especifique ese atributo aquí.
- Clave de roles: si desea utilizar roles de backend (se recomienda), especifique un atributo de la aserción en este campo; por ejemplo, `role` o `group`. Esta es otra situación en la que herramientas como Rastreo SAML pueden ayudar.
- Tiempo de vida de la sesión: de manera predeterminada, la sesión de los usuarios de OpenSearch Dashboards se cierra al cabo de 24 horas. Puede establecer este valor en cualquier número comprendido entre 60 y 1440 (24 horas) especificando un nuevo valor.

Cuando la configuración le parezca adecuada, guarde el dominio.

Paso 6: Actualizar las URL del IdP

Si ha habilitado la autenticación SAML mientras creaba un dominio, habrá tenido que especificar URL temporales para el IdP con el fin de generar el archivo de metadatos XML. Cuando el estado del dominio cambia a `Active`, puede obtener las URL correctas y modificar el IdP.

Para recuperar las URL, seleccione el dominio, y luego Acciones y Editar la configuración de seguridad. En Autenticación SAML para OpenSearch Dashboards/Kibana, puede encontrar el ID de entidad del proveedor de servicios y las URL de SSO correctos. Copie los valores y utilícelos para configurar el proveedor de identidades, sustituyendo las URL temporales que proporcionó en el paso 2.

Paso 7: Asignar usuarios de SAML a roles

Una vez que el estado del dominio sea Active (Activo) y el IdP esté configurado correctamente, vaya a OpenSearch Dashboards.

- Si eligió la dirección URL iniciada por el SP, diríjase a *domain-endpoint*/_dashboards. Para iniciar sesión directamente en un inquilino específico, puede agregar ?security_tenant=*tenant-name* a la URL.
- Si eligió la dirección URL iniciada por el IdP, diríjase al directorio de aplicaciones de su proveedor de identidad.

En ambos casos, inicie sesión como usuario maestro SAML o como usuario que pertenece al rol de backend maestro SAML. Para continuar con el ejemplo del paso 7, inicie sesión como jdoe o un miembro del grupo admins.

Una vez que OpenSearch Dashboards se cargue, seleccione Seguridad, Roles. Luego, asigne roles para permitir que otros usuarios accedan a OpenSearch Dashboards.

Por ejemplo, podría asignar a su colega de confianza jroe a los roles all_access y security_manager. También puede asignar el rol de backend analysts a los roles readall y opensearch_dashboards_user.

Si prefiere utilizar la API en lugar de OpenSearch Dashboards, consulte el siguiente ejemplo de solicitud:

```
PATCH _plugins/_security/api/rolesmapping
[
  {
    "op": "add", "path": "/security_manager", "value": { "users": ["master-user",
 "jdoe", "jroe"], "backend_roles": ["admins"] }
  },
  {
    "op": "add", "path": "/all_access", "value": { "users": ["master-user", "jdoe",
 "jroe"], "backend_roles": ["admins"] }
```

```
  },
  {
    "op": "add", "path": "/readall", "value": { "backend_roles": ["analysts"] }
  },
  {
    "op": "add", "path": "/opensearch_dashboards_user", "value": { "backend_roles":
 ["analysts"] }
  }
]
```

Configuración de la autenticación iniciada por proveedor de servicios y por proveedor de identidades

Si desea configurar la autenticación iniciada por SP e IdP, debe hacerlo a través de su proveedor de identidades. Por ejemplo, en Okta, puede seguir estos pasos:

1. Dentro de su aplicación SAML, vaya a General, Configuración SAML.
2. Para la URL de inicio de sesión único, proporcione URL SSO iniciado por IdP. Por ejemplo, `https://search-`*`domain-hash`*`/_dashboards/_opendistro/_security/saml/acs/`*`idpinitiated`*.
3. Habilite Permitir que esta aplicación solicite otras URL de SSO.
4. En Direcciones URL de SSO que se pueden solicitar, agregue una o varias URL de SSO iniciadas por SP. Por ejemplo, `https://search-`*`domain-hash`*`/_dashboards/_opendistro/_security/saml/`*`acs`*.

Configuración de la autenticación SAML (AWS CLI)

El siguiente comando de la AWS CLI habilita la autenticación SAML para OpenSearch Dashboards en un dominio existente:

```
aws opensearch update-domain-config \
  --domain-name my-domain \
  --advanced-security-options '{"SAMLOptions":{"Enabled":true,"MasterUserName":"my-
idp-user","MasterBackendRole":"my-idp-group-or-role","Idp":{"EntityId":"entity-
id","MetadataContent":"metadata-content-with-quotes-escaped"},"RolesKey":"optional-
roles-key","SessionTimeoutMinutes":180,"SubjectKey":"optional-subject-key"}}'
```

Debe escapar de todas las comillas y caracteres de nueva línea en el XML de metadatos. Por ejemplo, utilice `<KeyDescriptor use=\"signing\">\n` en lugar de `<KeyDescriptor use="signing">` y un salto de línea. Para obtener información detallada sobre la utilización de la AWS CLI, consulte la Referencia de los comandos de la AWS CLI.

Configuración de la autenticación SAML (API de configuración)

El siguiente comando de la API de configuración habilita la autenticación SAML para OpenSearch Dashboards en un dominio existente:

```
POST https://es.us-east-1.amazonaws.com/2021-01-01/opensearch/domain/my-domain/config
{
  "AdvancedSecurityOptions": {
    "SAMLOptions": {
      "Enabled": true,
      "MasterUserName": "my-idp-user",
      "MasterBackendRole": "my-idp-group-or-role",
      "Idp": {
        "EntityId": "entity-id",
        "MetadataContent": "metadata-content-with-quotes-escaped"
      },
      "RolesKey": "optional-roles-key",
      "SessionTimeoutMinutes": 180,
      "SubjectKey": "optional-subject-key"
    }
  }
}
```

Debe escapar de todas las comillas y caracteres de nueva línea en el XML de metadatos. Por ejemplo, utilice `<KeyDescriptor use=\"signing\">\n` en lugar de `<KeyDescriptor use="signing">` y un salto de línea. Para obtener información detallada sobre la utilización de la API de configuración, consulte la Referencia de la API de OpenSearch Service.

Solución de problemas de SAML

Error	Detalles
Your request: "*/some/path* " is not allowed (Su solicitud, "/some/path", no está permitida).	Compruebe que ha proporcionado la dirección URL de SSO correcta (paso 3) a su proveedor de identidad .

Error	Detalles
Proporcione un documento de metadatos del proveedor de identidad válido para habilitar SAML.	El archivo de metadatos del IdP no cumple con el estándar SAML 2.0. Verifique si hay errores mediante una herramienta de validación.
Las opciones de configuración de SAML no son visibles en la consola.	Actualice a la versión más reciente del software de servicio.
Error de configuración de SAML: se produjo un error al recuperar la configuración de SAML. Verifique su configuración.	Este error genérico puede producirse por muchas razones. • Compruebe que ha proporcionado a su proveedor de identidad el ID de entidad del SP y la dirección URL de SSO correctos. • Vuelva a generar el archivo de metadatos del IdP y verifique el ID de entidad de IdP. Agregue todos los metadatos actualizados en la consola de AWS. • Compruebe que la política de acceso a dominios permita obtener acceso a OpenSearch Dashboards y `_plugins/_security/*` . En general, recomendamos una política de acceso abierto para dominios que utilicen un control de acceso detallado. • Consulte la documentación de su proveedor de identidades a fin de conocer los pasos necesarios para configurar SAML.

Error	Detalles
Rol faltante: no hay roles disponibles para este usuario. Póngase en contacto con el administrador del sistema.	Se ha autenticado correctamente, pero el nombre de usuario y los roles de backend de la aserción SAML no se asignan a ningún rol y, por lo tanto, no tienen permisos. Estos mapeos distinguen entre mayúsculas y minúsculas. El administrador del sistema puede verificar el contenido de su aserción SAML con una herramienta como SAML-tracer y comprobar la asignación de roles con la siguiente solicitud: `GET _plugins/_security/api/rolesmapping`
Su navegador redirige continuamente o recibe errores HTTP 500 al intentar acceder a OpenSearch Dashboards.	Estos errores pueden producirse si la aserción SAML contiene un gran número de roles con un total aproximado de 1500 caracteres. Por ejemplo, si transfiere 80 roles, cuya longitud media es de 20 caracteres, es posible que supere el límite de tamaño de las cookies en su navegador web. A partir de la versión 2.7 de OpenSearch, la aserción SAML admite roles de hasta 5000 caracteres.
No puede cerrar sesión en ADFS.	ADFS requiere que se firme toda la solicitud de cierre de sesión, lo que OpenSearch Service no admite. Elimine `<SingleLogoutService />` del archivo de metadatos del IdP para forzar a OpenSearch Service a utilizar su propio mecanismo de cierre de sesión interno.

Error	Detalles
`Could not find entity descriptor for __PATH__.`	El ID de entidad del IdP proporcionado en el XML de metadatos a OpenSearch Service es diferente al de la respuesta de SAML. Para ello, asegúrese de que coincidan. Active los registros de errores de aplicacio nes de CW en su dominio para encontrar el mensaje de error y depurar el problema de integración con SAML.
`Signature validation failed. SAML response rejected.`	OpenSearch Service no puede verificar la firma de la respuesta de SAML mediante el certificado del IdP proporcionado en el XML de metadatos. Puede deberse a un error manual o a que su IdP haya rotado su certificado. Actualice el certificado más reciente de su IdP en el XML de metadatos proporcio nado a OpenSearch Service a través del Consola de administración de AWS.
`__PATH__ is not a valid audience for this response.`	El campo de audiencia de la respuesta de SAML no coincide con el punto de conexión del dominio. Para corregir este error, actualice el campo de audiencia del SP para que coincida con el punto de conexión de su dominio. Si ha activado los puntos de conexión personalizados, el campo de audiencia debe coincidir con su punto de conexión personalizado. Active los registros de errores de aplicaciones de CW en su dominio para encontrar el mensaje de error y depurar el problema de integración con SAML.
Su navegador recibe un mensaje de error HTTP 400 con `Invalid Request Id` en la respuesta.	Este error suele producirse si ha configurado la URL iniciada por el IdP con este formato *<DashboardsURL>* `/_opendistro/_security/saml/acs`. En su lugar, configure la URL con el formato *<DashboardsURL>* `/_opendistro/_security/saml/acs/idpinitiated` .

Error	Detalles
La respuesta se recibió en `__PATH__` en vez de `__PATH__`.	El campo de destino de la respuesta SAML no coincide con ninguno de los siguientes formatos de URL: • *<DashboardsURL>* `/_opendistro/_security/saml/acs` • *<DashboardsURL>* `/_opendistro/_security/saml/acs/idpinitiated` . Según el flujo de inicio de sesión que utilice (iniciado por SP o iniciado por IDP), introduzca un campo de destino que coincida con una de las URL de OpenSearch.
La respuesta tiene un atributo `InResponseTo` , pero no se esperaba `InResponseTo` .	Está utilizando la URL iniciada por IdP para un flujo de inicio de sesión iniciado por SP. En su lugar, utilice la URL iniciada por SP.

Deshabilitar la autenticación SAML

Para deshabilitar la autenticación SAML en OpenSearch Dashboards (consola)

1. Seleccione el dominio, Acciones y Editar la configuración de seguridad.
2. Desmarque Habilitar la autenticación SAML.
3. Seleccione Guardar cambios.
4. Después de que el dominio termine de procesar, compruebe el mapeo de roles de control de acceso detallado con la siguiente solicitud:

   ```
   GET _plugins/_security/api/rolesmapping
   ```

   Deshabilitar la autenticación SAML para Dashboards no elimina los mapeos del nombre de usuario maestro SAML o del rol de backend maestro SAML. Si desea eliminar estos mapeos, inicie sesión en Dashboards con la base de datos de usuario interna (si está habilitada) o utilice la API para eliminarlos:

```
PUT _plugins/_security/api/rolesmapping/all_access
{
  "users": [
    "master-user"
  ]
}
```

Compatibilidad de la propagación de identidades de confianza del Centro de identidades de IAM con OpenSearch

Ahora puede utilizar las entidades principales de Centro de identidades de IAM de AWS (usuarios y grupos) configuradas de forma centralizada a través de Propagación de identidades de confianza para acceder a los dominios de Amazon OpenSearch Service mediante las aplicaciones de OpenSearch Service. Para habilitar la compatibilidad del Centro de identidades de IAM con OpenSearch, necesitará habilitar el uso del Centro de identidades de IAM. Para obtener más información sobre cómo hacerlo, consulte ¿Qué es el Centro de identidades de IAM?. Consulte ¿Cómo asociar el dominio de OpenSearch como origen de datos en las aplicaciones de OpenSearch? para obtener más información.

Puede configurar el Centro de identidades de IAM mediante la consola de OpenSearch Service, la AWS Command Line Interface (AWS CLI) o los SDK de AWS.

> **Note**
>
> Dashboards (ubicado junto con el clúster) no admite las entidades principales del Centro de identidades de IAM. Solo se admiten a través de la interfaz de usuario centralizada de OpenSearch (Dashboards).

Consideraciones

Antes de utilizar Centro de identidades de IAM con Amazon OpenSearch Service, debe tener en cuenta lo siguiente:

- El Centro de identidades de IAM debe estar habilitado en la cuenta.
- El Centro de identidades de IAM debe estar disponible en su región.

- El dominio de OpenSearch tiene que tener la versión 1.3 o una posterior.
- El Control de acceso detallado debe estar habilitado en el dominio.
- El dominio debe estar en la misma región que la instancia del Centro de identidades de IAM.
- El dominio y la aplicación de OpenSearch deben pertenecer a la misma cuenta de AWS.

Modificación de la política de acceso al dominio

Antes de configurar el Centro de identidades de IAM, debe actualizar la política de acceso al dominio o los permisos del rol de IAM configurado en las aplicaciones de OpenSearch para la propagación de identidades de confianza.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:role/OpenSearchRole"
            },
            "Action": "es:ESHttp*",
            "Resource": "arn:aws:es:us-east-1:111122223333:domain/example-domain/
*"
        }
    ]
}
```

Configuración de la autenticación y autorización de Centro de identidades de IAM (consola)

Puede habilitar la autenticación y la autorización del Centro de identidades de IAM durante el proceso de creación del dominio o mediante la actualización de un dominio existente. Los pasos de configuración varían ligeramente según la opción que seleccione.

En los siguientes pasos, se explica cómo configurar un dominio existente para la autenticación y la autorización del Centro de identidades de IAM en la consola de Amazon OpenSearch Service:

1. En Configuración del dominio, vaya a Configuración de seguridad, seleccione Editar, vaya a la sección de autenticación del Centro de identidades de IAM y seleccione Habilitar el acceso a la API autenticado con el Centro de identidades de IAM.
2. Seleccione las claves SubjectKey y Roles de la siguiente manera.
   - Clave de asunto: elija una de las siguientes opciones: UserId (predeterminada), UserName e Email a fin de utilizar el atributo correspondiente como entidad principal para acceder al dominio.
   - Clave de roles: elija una de las siguientes opciones: GroupId (predeterminada) y GroupName para utilizar los valores de atributo correspondientes como rol de backend para un control de acceso detallado destinado a todos los grupos asociados a la entidad principal de IdC.

Una vez hechos los cambios, guarde el dominio.

Configuración del control de acceso detallado

Una vez que haya activado la opción Centro de identidades de IAM en su dominio de OpenSearch, podrá configurar el acceso a las entidades principales del Centro de identidades de IAM mediante la creación de una asignación de roles con el rol de backend. El valor del rol de backend de la entidad principal se basa en la pertenencia al grupo de la entidad principal de IdC y en la configuración de RolesKey de GroupId o GroupName.

> **Note**
>
> Amazon OpenSearch Service puede admitir hasta 100 grupos para un solo usuario. Si intenta utilizar más instancias de las permitidas, experimentará una incoherencia en el procesamiento detallado de las autorizaciones de control de acceso y recibirá un mensaje de error 403.

Configuración de la autenticación y autorización del Centro de identidades de IAM (CLI)

```
aws opensearch update-domain-config \
    --domain-name my-domain \
```

```
      --identity-center-options '{"EnabledAPIAccess": true,
 "IdentityCenterInstanceARN": "instance arn",  "SubjectKey": "UserId/UserName/
UserEmail" , "RolesKey": "GroupId/GroupName"}'
```

Deshabilitación de la autenticación del Centro de identidades de IAM en el dominio

Para deshabilitar Centro de identidades de IAM en su dominio de OpenSearch:

1. Seleccione el dominio, Acciones y Editar la configuración de seguridad.
2. Desmarque Habilitar el acceso a la API autenticado con el Centro de identidades de IAM.
3. Elija Guardar cambios.
4. Una vez que termine el procesamiento del dominio, elimine las asignaciones de roles agregadas para las entidades principales de IdC.

Para deshabilitar el Centro de identidades de IAM a través de la CLI, puede utilizar lo siguiente:

```
aws opensearch update-domain-config \
    --domain-name my-domain \
    --identity-center-options '{"EnabledAPIAccess": false}'
```

Configuración de la autenticación de Amazon Cognito para OpenSearch Dashboards

Puede autenticar y proteger la instalación predeterminada de Amazon OpenSearch Service para OpenSearch Dashboards mediante Amazon Cognito. La autenticación de Amazon Cognito es opcional y solo está disponible para dominios que utilicen OpenSearch o Elasticsearch 5.1 o una versión posterior. Si no configura la autenticación de Amazon Cognito, puede seguir protegiendo Dashboards con una política de acceso basada en IP y un servidor proxy, una autenticación básica de HTTP o SAML.

Gran parte del proceso de autenticación se produce en Amazon Cognito, pero esta sección ofrece instrucciones y requisitos para configurar los recursos de Amazon Cognito para trabajar con dominios de OpenSearch Service. Se aplican precios estándar a todos los recursos de Amazon Cognito.

> **Tip**
>
> Recomendamos utilizar la consola la primera vez que configure un dominio para utilizar la autenticación de Amazon Cognito para OpenSeach Dashboards. Los recursos de Amazon Cognito se pueden personalizar en gran medida y la consola puede ayudar a identificar y comprender las funciones importantes.

Temas

- Requisitos previos
- Configuración de un dominio para utilizar la autenticación de Amazon Cognito
- Permitir el rol autenticado
- Configuración de proveedores de identidades
- (Opcional) Configuración de acceso pormenorizado
- (Opcional) Personalización de la página de inicio de sesión
- (Opcional) Configuración de seguridad avanzada
- Testeo
- Cuotas
- Problemas habituales de configuración
- Deshabilitar la autenticación de Amazon Cognito para OpenSearch Dashboards
- Eliminación de dominios que utilizan la autenticación de Amazon Cognito para OpenSearch Dashboards

Requisitos previos

Antes de poder configurar la autenticación de Amazon Cognito para OpenSearch Dashboards, debe cumplir varios requisitos previos. La consola de OpenSearch Service ayuda a simplificar la creación de estos recursos, pero comprender la finalidad de cada recurso ayuda a la hora de configurar y resolver problemas. La autenticación de Amazon Cognito para Dashboards requiere los siguientes recursos:

- Grupo de usuarios de Amazon Cognito
- Grupo de identidades de Amazon Cognito

- Rol de IAM que tiene la política de `AmazonOpenSearchServiceCognitoAccess` adjunta (`CognitoAccessForAmazonOpenSearch`)

> **Note**
>
> El grupo de usuarios y el grupo de identidades deben estar en la misma Región de AWS. Puede utilizar el mismo grupo de usuarios, el mismo grupo de identidades y el mismo rol de IAM para agregar la autenticación de Amazon Cognito para Dashboards a varios dominios de OpenSearch Service. Para obtener más información, consulte the section called “Cuotas”.

Acerca del grupo de usuarios

Los grupos de usuarios tiene dos características principales: crear y administrar un directorio de usuarios y permitir que los usuarios se inscriban e inicien sesión. Para obtener instrucciones acerca de cómo crear un grupo de usuarios, consulte Introducción a los grupos de usuarios en la Guía para desarrolladores de Amazon Cognito.

Al crear un grupo de usuarios que se va a utilizar con OpenSearch Service, tenga en cuenta lo siguiente:

- El grupo de usuarios de Amazon Cognito debe tener un nombre de dominio. OpenSearch Service utiliza este nombre de dominio para redirigir a los usuarios a una página de inicio de sesión para acceder a Dashboards. Aparte de un nombre de dominio, el grupo de usuarios no requiere ninguna configuración no predeterminada.
- Debe especificar los atributos estándar necesarios del grupo como nombre, fecha de nacimiento, dirección de email y número de teléfono. No se pueden cambiar estos atributos después de crear el grupo de usuarios, por lo que debe elegir los que considere importantes en este momento.
- Al crear el grupo de usuarios, elija si los usuarios pueden crear sus propias cuentas, el nivel mínimo de seguridad de la contraseña para las cuentas y si se va a habilitar la autenticación multifactor. Si tiene previsto utilizar un proveedor de identidades externo, estos ajustes no tienen importancia. Técnicamente puede habilitar el grupo de usuarios como un proveedor de identidades y habilitar un proveedor de identidades externo, pero la mayoría de las personas prefieren uno u otro.

Los ID de grupos de usuarios adoptan la forma de *`region_ID`*. Si tiene previsto utilizar la CLI de AWS o un SDK de AWS para configurar OpenSearch Service, anote el ID.

Acerca del grupo de identidades

Los grupos de identidades permiten asignar funciones temporales con privilegios limitados a los usuarios después de que inicien sesión. Para obtener instrucciones acerca de cómo crear un grupo de identidades, consulte Información general de la consola de grupos de identidades en la Guía para desarrolladores de Amazon Cognito. Al crear un grupo de identidades que se va a utilizar con OpenSearch Service, tenga en cuenta lo siguiente:

- Si utiliza la consola de Amazon Cognito, debe seleccionar la casilla de verificación Habilitar el acceso a identidades sin autenticar para crear el grupo de identidades. Después de crear el grupo de identidades y de configurar el dominio de OpenSearch Service, Amazon Cognito deshabilita esta configuración.
- No es necesario agregar proveedores de identidades externos al grupo de identidades. Cuando se configura OpenSearch Service para utilizar la autenticación de Amazon Cognito, configura el grupo de identidades para utilizar el grupo de usuarios que acaba de crear.
- Después de crear el grupo de identidades, debe elegir funciones de IAM autenticadas y sin autenticar. Estos roles especifican las políticas de acceso que los usuarios tienen antes y después de que inicien sesión. Si utiliza la consola de Amazon Cognito, puede crear estas funciones para usted. Una vez creado el rol autenticado, anote el ARN, que adopta la forma de `arn:aws:iam::123456789012:role/Cognito_identitypoolnameAuth_Role`.

Los ID de grupos de identidades adoptan la forma de *region*:*ID*-*ID*-*ID*-*ID*-*ID*. Si tiene previsto utilizar la CLI de AWS o un SDK de AWS para configurar OpenSearch Service, anote el ID.

Acerca del rol CognitoAccessForAmazonOpenSearch

OpenSearch Service necesita permisos para configurar los grupos de usuarios y de identidades de Amazon Cognito y utilizarlos para la autenticación. Para esto, puede utilizar `AmazonOpenSearchServiceCognitoAccess`, que es una política administrada por AWS. `AmazonESCognitoAccess` es una política heredada que fue reemplazada por `AmazonOpenSearchServiceCognitoAccess` cuando el servicio fue renombrado a Amazon OpenSearch Service. Ambas políticas proporcionan los permisos mínimos de Amazon Cognito necesarios para habilitar la autenticación de Amazon Cognito. Para ver los detalles de las políticas, consulte AmazonOpenSearchServiceCognitoAccess en la Guía de referencia de políticas administradas de AWS.

Si utiliza la consola para crear o configurar el dominio de OpenSearch Service, se crea un rol de IAM para usted y se adjunta a dicho rol la política `AmazonOpenSearchServiceCognitoAccess` (o la

política `AmazonESCognitoAccess` si es un dominio de Elasticsearch). El nombre predeterminado del rol es `CognitoAccessForAmazonOpenSearch`.

Las políticas de permisos de rol `AmazonOpenSearchServiceCognitoAccess` y `AmazonESCognitoAccess` permiten a OpenSearch Service completar las siguientes acciones en todos los grupos de identidades y usuarios:

- Acción: `cognito-idp:DescribeUserPool`
- Acción: `cognito-idp:CreateUserPoolClient`
- Acción: `cognito-idp:DeleteUserPoolClient`
- Acción: `cognito-idp:UpdateUserPoolClient`
- Acción: `cognito-idp:DescribeUserPoolClient`
- Acción: `cognito-idp:AdminInitiateAuth`
- Acción: `cognito-idp:AdminUserGlobalSignOut`
- Acción: `cognito-idp:ListUserPoolClients`
- Acción: `cognito-identity:DescribeIdentityPool`
- Acción: `cognito-identity:SetIdentityPoolRoles`
- Acción: `cognito-identity:GetIdentityPoolRoles`

Si utiliza la AWS CLI o uno de los SDK de AWS, debe crear un rol propio, adjuntar la política y especificar el ARN para este rol al configurar el dominio de OpenSearch Service. El rol debe tener la siguiente relación de confianza:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "opensearchservice.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
```

```
}
```

Para obtener instrucciones, consulte Crear un rol para delegar permisos a un servicio de AWS y Adición y eliminación de permisos de identidad de IAM en la Guía del usuario de IAM.

Configuración de un dominio para utilizar la autenticación de Amazon Cognito

Después de completar los requisitos previos, puede configurar un dominio de OpenSearch Service para que utilice Amazon Cognito para Dashboards.

> **Note**
>
> Amazon Cognito no está disponible en todas las Regiones de AWS. Para ver una lista de las regiones admitidas, consulte Puntos de conexión del servicio de Amazon Cognito. No es necesario utilizar la misma región para Amazon Cognito que utiliza para OpenSearch Service.

Configuración de la autenticación de Amazon Cognito (consola)

Dado que se crea el rol `CognitoAccessForAmazonOpenSearch` para que usted no tenga que hacerlo, la consola ofrece la experiencia de configuración más sencilla. Además de los permisos de OpenSearch Service estándar, se necesita el siguiente conjunto de permisos para utilizar la consola con el fin de crear un dominio que emplee la autenticación de Amazon Cognito para OpenSearch Dashboards.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [{
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeVpcs",
        "cognito-identity:ListIdentityPools",
        "cognito-idp:ListUserPools",
        "iam:CreateRole",
```

```
        "iam:AttachRolePolicy"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "iam:GetRole",
        "iam:PassRole"
      ],
      "Resource": "arn:aws:iam::123456789012:role/service-
role/CognitoAccessForAmazonOpenSearch"
    }
  ]
}
```

Para obtener instrucciones sobre cómo agregar permisos a una identidad (usuario, grupo de usuarios o rol), consulteAdición de permisos de identidad de IAM (consola).

Si el `CognitoAccessForAmazonOpenSearch` ya existe, necesita menos permisos:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [{
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeVpcs",
        "cognito-identity:ListIdentityPools",
        "cognito-idp:ListUserPools"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "iam:GetRole",
        "iam:PassRole"
      ],
      "Resource": "arn:aws:iam::123456789012:role/service-
role/CognitoAccessForAmazonOpenSearch"
```

```
    }
  ]
}
```

Para configurar la autenticación de Amazon Cognito para Dashboards (consola)

1. Abra la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home/.
2. En Dominios, seleccione el dominio que desea configurar.
3. Seleccione Acciones y Editar la configuración de seguridad.
4. Seleccione Habilitar la autenticación de Amazon Cognito.
5. En Región, seleccione la Región de AWS que contiene el grupo de usuarios y de identidades de Amazon Cognito.
6. En Grupo de usuarios de Cognito, seleccione un grupo de usuarios o cree uno. Para obtener más información, consulte the section called “Acerca del grupo de usuarios”.
7. En Grupo de identidades de Cognito, seleccione un grupo de identidades o cree uno. Para obtener más información, consulte the section called “Acerca del grupo de identidades”.

   > **Note**
   >
   > Los enlaces Crear grupo de usuarios y Crear grupo de identidades dirigen a la consola de Amazon Cognito y requieren crear estos recursos de forma manual. El proceso no es automático. Para obtener más información, consulte the section called “Requisitos previos”.

8. Para Nombre de rol de IAM, utilice el valor de `CognitoAccessForAmazonOpenSearch` predeterminado (recomendado) o ingrese un nombre nuevo. Para obtener más información, consulte the section called “Acerca del rol CognitoAccessForAmazonOpenSearch”.
9. Elija Guardar cambios.

Después de que el dominio termine de procesarse, consulte the section called “Permitir el rol autenticado” y the section called “Configuración de proveedores de identidades” para conocer los pasos adicionales de configuración.

Configuración de la autenticación de Amazon Cognito (AWS CLI)

Utilice el parámetro `--cognito-options` para configurar el dominio de OpenSearch Service. Los comandos `create-domain` y `update-domain-config` emplean la siguiente sintaxis:

```
--cognito-options Enabled=true,UserPoolId="user-pool-id",IdentityPoolId="identity-pool-id",RoleArn="arn:aws:iam::123456789012:role/CognitoAccessForAmazonOpenSearch"
```

Ejemplo

En el siguiente ejemplo se crea un dominio en la región `us-east-1` que habilita la autenticación de Amazon Cognito para Dashboards mediante el rol `CognitoAccessForAmazonOpenSearch` y proporciona acceso al dominio a `Cognito_Auth_Role`:

```
aws opensearch create-domain --domain-name my-domain --
region us-east-1 --access-policies '{ "Version": "2012-10-17",
        "Statement":[{"Effect":"Allow","Principal":{"AWS":
 ["arn:aws:iam::123456789012:role/
Cognito_Auth_Role"]},"Action":"es:ESHttp*","Resource":"arn:aws:es:us-
east-1:123456789012:domain/*" }]}' --engine-version "OpenSearch_1.0"
 --cluster-config InstanceType=m4.xlarge.search,InstanceCount=1
 --ebs-options EBSEnabled=true,VolumeSize=10 --cognito-options
 Enabled=true,UserPoolId="us-east-1_123456789",IdentityPoolId="us-
east-1:12345678-1234-1234-1234-123456789012",RoleArn="arn:aws:iam::123456789012:role/
CognitoAccessForAmazonOpenSearch"
```

Después de que el dominio termine de procesarse, consulte the section called “Permitir el rol autenticado” y the section called “Configuración de proveedores de identidades” para conocer los pasos adicionales de configuración.

Configuración de la autenticación de Amazon Cognito (SDK de AWS)

Los AWS SDK (excepto los SDK de Android e iOS) admiten todas las operaciones definidas en la referencia de API de Amazon OpenSearch Service, incluido el parámetro `CognitoOptions` para las operaciones `CreateDomain` y `UpdateDomainConfig`. Para obtener más información acerca de cómo instalar y utilizar los SDK de AWS, consulte los Kits de desarrollo de software de AWS.

Después de que el dominio termine de procesarse, consulte the section called “Permitir el rol autenticado” y the section called “Configuración de proveedores de identidades” para conocer los pasos adicionales de configuración.

Permitir el rol autenticado

De forma predeterminada, el rol de IAM autenticado que configuró de acuerdo con las instrucciones de the section called “Acerca del grupo de identidades” no cuenta con los privilegios necesarios para obtener acceso a OpenSearch Dashboards. Debe proporcionar permisos adicionales al rol.

> **Note**
>
> Si configuró el control de acceso detallado y utiliza una política de acceso abierta o basada en IP, puede omitir este paso.

Puede incluir estos permisos en una política basada en identidades, pero, a menos que quiera que los usuarios autenticados obtengan acceso a todos los dominios de OpenSearch Service, el mejor enfoque es una política basada en recursos adjunta a un solo dominio.

Para el `Principal`, especifique el ARN del rol autenticado de Cognito que configuró con las pautas que figuran en the section called “Acerca del grupo de identidades”.

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Principal":{
            "AWS":[
               "arn:aws:iam::111122223333:role/Cognito_identitypoolname/
Auth_Role"
            ]
         },
         "Action":[
            "es:ESHttp*"
         ],
         "Resource":"arn:aws:es:us-east-1:111122223333:domain/domain-name/*"
      }
   ]
}
```

Para obtener instrucciones sobre cómo agregar una política basada en recursos a un dominio de OpenSearch Service, consulte the section called “Configuración de políticas de acceso”.

Configuración de proveedores de identidades

Al configurar un dominio para que utilice la autenticación de Amazon Cognito para Dashboards, OpenSearch Service agrega un cliente de aplicación al grupo de usuarios y agrega el grupo de usuarios al grupo de identidades como proveedor de autenticación.

> **Warning**
>
> No cambie el nombre ni elimine el cliente de aplicación.

En función de cómo configure el grupo de usuarios, puede que necesite crear cuentas de usuarios de forma manual, o bien los usuarios podrían crear sus propias cuentas. Si estos ajustes son aceptables, no es necesario realizar más acciones. Sin embargo, muchas personas prefieren usar proveedores de identidades externos.

Para habilitar un proveedor de identidades SAML 2.0, debe proporcionar un documento de metadatos de SAML. Para habilitar proveedores de identidades sociales como Login with Amazon, Facebook y Google, debe disponer de un ID de aplicación y de la clave secreta de la aplicación de esos proveedores. Puede habilitar cualquier combinación de proveedores de identidades.

La forma más sencilla de configurar el grupo de usuarios consiste en utilizar la consola de Amazon Cognito. Para obtener instrucciones, consulte Inicio de sesión en grupos de usuarios con proveedores de identidad de terceros y Ajustes específicos de una aplicación en los clientes de aplicación en la Guía para desarrolladores de Amazon Cognito.

(Opcional) Configuración de acceso pormenorizado

Es posible que haya observado que la configuración del grupo de identidades predeterminado asigna el mismo rol de IAM (`Cognito_`*`identitypool`*`Auth_Role`) a cada usuario que inicia sesión, lo cual significa que todos los usuarios pueden obtener acceso a los mismos recursos de AWS. Si desea utilizar el control de acceso detallado con Amazon Cognito, por ejemplo, si desea que los analistas de la organización tengan acceso de solo lectura a varios índices, pero que los desarrolladores tengan acceso de escritura a todos los índices, tiene dos opciones:

- Cree grupos de usuarios y configure su proveedor de identidades de tal forma que elija el rol de IAM en función del token de autenticación del usuario (recomendado).

- Configure su proveedor de identidades para que elija el rol de IAM en función de una o varias reglas.

Para obtener un tutorial que incluya el control de acceso detallado, consulte the section called "Tutorial: Control de acceso detallado con la autenticación de Cognito".

> **Important**
>
> Al igual que sucede con el rol predeterminado, Amazon Cognito debe formar parte de la relación de confianza de cada rol adicional. Para obtener más información, consulte Creación de roles para la asignación de roles en la Guía para desarrolladores de Amazon Cognito.

Grupos de usuarios y tokens

Al crear un grupo de usuarios, elige un rol de IAM para los miembros del grupo. Para obtener información acerca de cómo crear grupos, consulte Agregar grupos a un grupo de usuarios en la Guía para desarrolladores de Amazon Cognito.

Después de crear uno o más grupos de usuarios, puede configurar su proveedor de autenticación para asignar a los usuarios las funciones de sus grupos en lugar del rol predeterminado del grupo de identidades. Seleccione Elegir rol a partir de un token, luego elija Utilizar rol autenticado predeterminado o DENEGAR para especificar de qué manera el grupo de identidades se encargará de los usuarios que no forman parte del grupo.

Reglas

Las reglas son básicamente una serie de instrucciones `if` que Amazon Cognito evalúa de forma secuencial. Por ejemplo, si una dirección de email del usuario contiene `@corporate`, Amazon Cognito asigna `Role_A` a ese usuario. Si la dirección de correo electrónico de un usuario contiene `@subsidiary`, asigna a ese usuario `Role_B`. De lo contrario, asigna al usuario el rol autenticado predeterminado.

Para obtener más información, consulte Uso de la asignación basada en reglas para la asignación de roles a los usuarios en la Guía para desarrolladores de Amazon Cognito.

(Opcional) Personalización de la página de inicio de sesión

Puede usar la consola de Amazon Cognito para cargar un logotipo personalizado y realizar cambios de CSS en la página de inicio de sesión. Para obtener instrucciones y una lista completa de las propiedades de CSS, consulte Personalización de la marca de la interfaz de usuario alojada (clásica) en la Guía para desarrolladores de Amazon Cognito.

(Opcional) Configuración de seguridad avanzada

Los grupos de usuarios de Amazon Cognito admiten características de seguridad avanzadas como la autenticación multifactor, la verificación de credenciales comprometidas y la autenticación flexible. Para obtener más información, consulte Uso de las características de seguridad de los grupos de usuarios de Amazon Cognito en la Guía para desarrolladores de Amazon Cognito.

Testeo

Una vez que esté satisfecho con la configuración, verifique que la experiencia del usuario cumpla sus expectativas.

Para acceder a OpenSearch Dashboards

1. Vaya a `https://`*`opensearch-domain`*`/_dashboards` en un navegador Web. Para iniciar sesión directamente en un inquilino específico, agregue `?security_tenant=`*`tenant-name`* a la URL.
2. Inicie sesión con las credenciales que prefiera.
3. Una vez que se carga OpenSearch Dashboards, configure al menos un patrón de índice. Dashboards utiliza estos patrones para identificar los índices que desea analizar. Escriba `*`, seleccione Siguiente paso y, a continuación, seleccione Crear patrón de índice.
4. Para buscar datos o explorar en ellos, seleccione Detección.

Si cualquier paso de este proceso da error, consulte the section called “Problemas habituales de configuración” para obtener información sobre resolución de problemas.

Cuotas

Amazon Cognito cuenta con límites flexibles en muchos de los recursos. Si desea habilitar la autenticación de Dashboards para una gran cantidad de dominios de OpenSearch Service, consulte Cuotas en Amazon Cognito y solicite un aumento de los límites si es necesario.

Cada dominio de OpenSearch Service agrega un cliente de aplicación al grupo de usuarios, lo cual agrega un proveedor de autenticación al grupo de identidades. Si habilita la autenticación de OpenSearch Dashboards para más de 10 dominios, puede que llegue al límite de "máximo de proveedores de grupo de usuarios de Amazon Cognito por grupo de identidades". Si supera un límite, cualquier dominio de OpenSearch Service que intente configurar para que utilice la autenticación de Amazon Cognito para Dashboards puede bloquearse en un estado de configuración o En proceso.

Problemas habituales de configuración

En las tablas siguientes se muestran los problemas habituales de configuración y las soluciones.

Configuración de OpenSearch Service

Problema	Solución
`OpenSearch Service can't create the role` (consola)	No dispone de los permisos de IAM correctos. Agregue los permisos especificados en the section called "Configuración de la autenticación de Amazon Cognito (consola)".
`User is not authorized to perform: iam:PassRole on resource CognitoAccessForAmazonOpenSearch` (consola)	No cuenta con permisos `iam:PassRole` para el rol CognitoAccessForAmazonOpenSearch. Asocie la siguiente política a su cuenta: JSON `{ "Version":"2012-10-17", "Statement": [ { "Effect": "Allow", "Action": [ "iam:PassRole" ], "Resource": "arn:aws:iam:: 123456789012:role/service-role/CognitoAccessForAmazonOpenSearch " } ]`

Problema	Solución
	<pre>}</pre> Como alternativa, puede adjuntar la política `IAMFullAccess` .
`User is not authorized to perform: cognito-identity:ListIdentityPools on resource`	No tiene permisos de lectura para Amazon Cognito. Adjunte la política `AmazonCognitoReadOnly` a su cuenta.
`An error occurred (ValidationException) when calling the CreateDomain operation: OpenSearch Service must be allowed to use the passed role`	OpenSearch Service no se especifica en la relación de confianza del rol `CognitoAccessForAmazonOpenSearch` . Verifique si el rol utiliza la relación de confianza que se especifica en the section called “Acerca del rol CognitoAccessForAmazonOpenSearch”. De manera alternativa, utilice la consola para configurar la autenticación de Amazon Cognito. La consola crea un rol para usted.
`An error occurred (ValidationException) when calling the CreateDomain operation: User is not authorized to perform: cognito-idp: action on resource: user pool`	El rol especificado en `--cognito-options` no cuenta con permisos para acceder a Amazon Cognito. Verifique que el rol tenga adjunta la política administrada AWS de `AmazonOpenSearchServiceCognitoAccess` . De manera alternativa, utilice la consola para configurar la autenticación de Amazon Cognito. La consola crea un rol para usted.
`An error occurred (ValidationException) when calling the CreateDomain operation: User pool does not exist`	OpenSearch Service no encuentra el grupo de usuarios. Verifique que creó uno y que tiene el ID correcto. Para encontrar el ID, puede utilizar la consola de Amazon Cognito o el siguiente comando de la AWS CLI: <pre>aws cognito-idp list-user-pools --max-results 60 --region *region*</pre>

Problema	Solución
`An error occurred (ValidationException) when calling the CreateDomain operation: IdentityPool not found`	OpenSearch Service no encuentra el grupo de identidades. Verifique que creó uno y que tiene el ID correcto. Para encontrar el ID, puede utilizar la consola de Amazon Cognito o el siguiente comando de la AWS CLI: `aws cognito-identity list-identity-pools --max-results 60 --region region`
`An error occurred (ValidationException) when calling the CreateDomain operation: Domain needs to be specified for user pool`	El grupo de usuarios no tiene un nombre de dominio. Puede configurar uno con la consola de Amazon Cognito o el siguiente comando de la AWS CLI: `aws cognito-idp create-user-pool-domain --domain name --user-pool-id id`

Acceso a OpenSearch Dashboards

Problema	Solución
La página de inicio de sesión no muestra mis proveedores de identidades preferidos.	Verifique que habilitó el proveedor de identidades del cliente de la aplicación de OpenSearch Service tal y como se especifica en the section called “Configuración de proveedores de identidades”.
La página de inicio de sesión no parece estar asociada a mi organización.	Consulte the section called “(Opcional) Personalización de la página de inicio de sesión”.
Mis credenciales de inicio de sesión no funcionan.	Verifique que configuró el proveedor de identidades tal y como se especifica en the section called “Configuración de proveedores de identidades”. Si utiliza el grupo de usuarios como proveedor de identidades, verifique que la cuenta exista en la consola de Amazon Cognito.

Problema	Solución
OpenSearch Dashboards no se carga en absoluto o no funciona correctamente.	El rol autenticado de Amazon Cognito necesita permisos `es:ESHttp*` para el dominio (`/*`) a fin de obtener acceso y utilizar Dashboards. Verifique que agregó una política de acceso tal y como se especifica en the section called “Permitir el rol autenticado”.
Cuando cierro sesión en OpenSearch Dashboard desde una pestaña, las pestañas restantes muestran un mensaje que indica que se ha revocado el token de actualización.	Al cerrar sesión en una sesión de OpenSearch Dashboards con la autenticación de Amazon Cognito, OpenSearch Service ejecuta la operación AdminUserGlobalSignOut, que cierra sesión en todas las sesiones activas de OpenSearch Dashboards.

Problema	Solución
`Invalid identity pool configuration. Check assigned IAM roles for this pool.`	Amazon Cognito no cuenta con permisos para asumir el rol de IAM en nombre del usuario autenticado. Modifique la relación de confianza para el rol de tal forma que incluya: JSON `{ "Version":"2012-10-17", "Statement": [{ "Effect": "Allow", "Principal": { "Federated": "cognito-identity.amazonaws.com" }, "Action": "sts:AssumeRoleWithWebIdentity", "Condition": { "StringEquals": { "cognito-identity.amazonaws.com:aud": " identity-pool-id " }, "ForAnyValue:StringLike": { "cognito-identity.amazonaws.com:amr": "authenticated" } } }] }`
`Token is not from a supported provider of this identity pool.`	Este error poco habitual se puede producir al eliminar el cliente de aplicación del grupo de usuarios. Intente abrir Dashboards en una nueva sesión del navegador.

Deshabilitar la autenticación de Amazon Cognito para OpenSearch Dashboards

Utilice el siguiente procedimiento para deshabilitar la autenticación de Amazon Cognito para Dashboards.

Para deshabilitar la autenticación de Amazon Cognito para Dashboards (consola)

1. Abra la consola de Amazon OpenSearch Service.
2. En Dominios, seleccione el dominio que desea configurar.
3. Seleccione Acciones y Editar la configuración de seguridad.
4. Desactive Habilitar la autenticación de Amazon Cognito.
5. Seleccione Guardar cambios.

> **Important**
>
> Si ya no necesita el grupo de usuarios y de identidades de Amazon Cognito, elimínelos. De lo contrario, seguirá incurriendo en costos.

Eliminación de dominios que utilizan la autenticación de Amazon Cognito para OpenSearch Dashboards

Para evitar que los dominios que utilizan la autenticación de Amazon Cognito para Dashboards se bloqueen en un estado de configuración En proceso, elimine los dominios de OpenSearch Service antes de eliminar los grupos de identidades y de usuarios de Amazon Cognito asociados.

Uso de roles vinculados a servicios para Amazon OpenSearch Service.

Amazon OpenSearch Service utiliza roles vinculados al servicio de AWS Identity and Access Management (IAM). Un rol vinculado a servicios es un tipo único de rol de IAM que está vinculado directamente a OpenSearch Service. OpenSearch Service predefine los roles vinculados a servicios y estos incluyen todos los permisos que el servicio requiere para llamar a otros servicios de AWS en su nombre.

Un rol vinculado a servicios simplifica la configuración de OpenSearch Service porque ya no tendrá que agregar manualmente los permisos necesarios. OpenSearch Service define los permisos de sus roles vinculados a servicios y, a menos que esté definido de otra manera, solo OpenSearch Service puede asumir sus roles. Los permisos definidos incluyen las políticas de confianza y de permisos, y que la política de permisos no se pueda asociar a ninguna otra entidad de IAM. Para obtener información actualizada sobre las políticas de permisos y roles vinculados a servicios, consulte Historial de documentos de Amazon OpenSearch Service.

Para obtener información sobre otros servicios que admiten roles vinculados al servicio, consulte Servicios de AWS que funcionan con IAM y busque aquellos para los que aparezca Sí en la columna Roles vinculados al servicio. Elija una opción Sí con un enlace para ver la documentación acerca del rol vinculado al servicio en cuestión.

Temas

- Uso de roles vinculados al servicio para crear dominios de VPC y orígenes de datos de consulta directa
- Uso de roles vinculados a un servicio para crear colecciones de OpenSearch sin servidor
- Uso de roles vinculados a servicio para crear canalizaciones de OpenSearch Ingestion

Uso de roles vinculados al servicio para crear dominios de VPC y orígenes de datos de consulta directa

Amazon OpenSearch Service utiliza roles vinculados al servicio de AWS Identity and Access Management (IAM). Un rol vinculado a servicios es un tipo único de rol de IAM que está vinculado directamente a OpenSearch Service. OpenSearch Service predefine los roles vinculados a servicios y estos incluyen todos los permisos que el servicio requiere para llamar a otros servicios de AWS en su nombre.

OpenSearch Service utiliza el rol vinculado al servicio denominado AWSServiceRoleForAmazonOpenSearchService, que proporciona los permisos mínimos de Amazon EC2 y Elastic Load Balancing necesarios para que el rol habilite el acceso de VPC para un dominio o un origen de datos de consulta directa.

Función heredada de Elasticsearch

Amazon OpenSearch Service utiliza un rol vinculado a servicios denominado `AWSServiceRoleForAmazonOpenSearchService`. Es posible que

las cuentas también contengan un rol vinculado a servicio denominado `AWSServiceRoleForAmazonElasticsearchService`, que funciona con los puntos de conexión de la API de Elasticsearch obsoletos.

Si el rol de Elasticsearch heredado no existe en la cuenta, OpenSearch Service creará automáticamente un nuevo rol vinculado a servicio de OpenSearch la primera vez que se cree un dominio de OpenSearch. En caso contrario, la cuenta seguirá utilizando el rol de Elasticsearch. Para que esta creación automática se realice correctamente, es necesario disponer de permisos para la acción `iam:CreateServiceLinkedRole`.

Permisos

El rol vinculado a servicios `AWSServiceRoleForAmazonOpenSearchService` confía en los siguientes servicios para asumir el rol:

- `opensearchservice.amazonaws.com`

La política de permisos del rol denominada `AmazonOpenSearchServiceRolePolicy` permite que OpenSearch Service realice las siguientes acciones en los recursos especificados:

- Acción: `acm:DescribeCertificate` en *
- Acción: `cloudwatch:PutMetricData` en *
- Acción: `ec2:CreateNetworkInterface` en *
- Acción: `ec2:DeleteNetworkInterface` en *
- Acción: `ec2:DescribeNetworkInterfaces` en *
- Acción: `ec2:ModifyNetworkInterfaceAttribute` en *
- Acción: `ec2:DescribeSecurityGroups` en *
- Acción: `ec2:DescribeSubnets` en *
- Acción: `ec2:DescribeVpcs` en *
- Acción: `ec2:CreateTags` en todas las interfaces de red y puntos de conexión de VPC
- Acción: `ec2:DescribeTags` en *
- Acción: `ec2:CreateVpcEndpoint` en todas las VPC, grupos de seguridad, subredes y tablas de enrutamiento, así como en todos los puntos de conexión de VPC, cuando la solicitud contiene la etiqueta `OpenSearchManaged=true`

- Acción: `ec2:ModifyVpcEndpoint` en todas las VPC, grupos de seguridad, subredes y tablas de enrutamiento, así como en todos los puntos de conexión de VPC, cuando la solicitud contiene la etiqueta `OpenSearchManaged=true`
- Acción: `ec2:DeleteVpcEndpoints` en todos los extremos cuando la solicitud contiene la etiqueta `OpenSearchManaged=true`
- Acción: `ec2:AssignIpv6Addresses` en *
- Acción: `ec2:UnAssignIpv6Addresses` en *
- Acción: `elasticloadbalancing:AddListenerCertificates` en *
- Acción: `elasticloadbalancing:RemoveListenerCertificates` en *

Debe configurar permisos para permitir a una entidad de IAM (como un usuario, grupo o rol) crear, editar o eliminar un rol vinculado a servicios. Para obtener más información, consulte Permisos de roles vinculados a servicios en la Guía del usuario de IAM.

Creación del rol vinculado a servicio

No necesita crear manualmente un rol vinculado a servicios. Cuando usted crea un dominio habilitado para VPC o un origen de datos de consulta directa mediante Consola de administración de AWS, OpenSearch Service crea el rol vinculado al servicio por usted. Para que esta creación automática se realice correctamente, es necesario disponer de permisos para la acción `iam:CreateServiceLinkedRole`.

También puede utilizar la consola de IAM, la CLI de IAM o la API de IAM para crear un rol vinculado a servicios manualmente. Para obtener más información, consulte Creating a service-linked role en la Guía del usuario de IAM.

Edición del rol vinculado al servicio

OpenSearch Service no permite editar el rol vinculado a servicios `AWSServiceRoleForAmazonOpenSearchService`. Después de crear un rol vinculado a un servicio, no puede cambiarle el nombre, ya que varias entidades pueden hacer referencia a él. Sin embargo, puede editar la descripción del rol mediante IAM. Para obtener más información, consulte Editar un rol vinculado a servicios en la Guía del usuario de IAM.

Eliminación del rol vinculado a un servicio

Si ya no necesita usar una característica o servicio que requieran un rol vinculado a un servicio, le recomendamos que elimine dicho rol. De esta forma, no tiene una entidad no utilizada que no se

monitoree ni mantenga de forma activa. Sin embargo, debe limpiar el rol vinculado a servicios antes de eliminarlo manualmente.

Limpieza del rol vinculado al servicio

Antes de poder utilizar IAM para eliminar un rol vinculado a un servicio, primero debe confirmar que dicho rol no tiene sesiones activas y eliminar los recursos que utiliza.

Para comprobar si el rol vinculado a un servicio tiene una sesión activa en la consola de IAM

1. Inicie sesión en la Consola de administración de AWS y abra la consola de IAM en https://console.aws.amazon.com/iam/.
2. En el panel de navegación de la consola de IAM, elija Roles (Roles). A continuación, seleccione el nombre (no la casilla de verificación) del rol de `AWSServiceRoleForAmazonOpenSearchService`.
3. En la página Resumen del rol seleccionado, seleccione la pestaña Asesor de acceso.
4. En la pestaña Asesor de acceso, revise la actividad reciente del rol vinculado a servicios.

> **Note**
>
> Si no está seguro de si OpenSearch Service utiliza el rol `AWSServiceRoleForAmazonOpenSearchService`, puede intentar eliminar el rol. Si el servicio utiliza el rol, este no podrá eliminarse y se podrán ver los recursos que lo utilizan. Si el rol se está utilizando, debe esperar que la sesión finalice para poder eliminarlo, o para eliminar los recursos que lo utilizan. No se puede revocar la sesión de un rol vinculado a servicios.

Eliminar manualmente un rol vinculado a servicios

Eliminar roles vinculados a servicios de la consola de IAM, API o la CLI de AWS. Para obtener más información, consulte Deleting a service-linked role en la Guía del usuario de IAM.

Uso de roles vinculados a un servicio para crear colecciones de OpenSearch sin servidor

OpenSearch sin servidor utiliza roles vinculados al servicio de AWS Identity and Access Management (IAM). Un rol vinculado a servicios es un tipo único de rol de IAM que está vinculado directamente a

OpenSearch Service. OpenSearch Service predefine los roles vinculados a servicios y estos incluyen todos los permisos que el servicio requiere para llamar a otros servicios de AWS en su nombre.

OpenSearch sin servidor usa el rol vinculado al servicio denominado AWSServiceRoleForAmazonOpenSearchServerless, que proporciona los permisos necesarios para que el rol publique las métricas de CloudWatch sin servidor en su cuenta. La política de permisos del rol asociada a AWSServiceRoleForAmazonOpenSearchServerless se llama `AmazonOpenSearchServerlessServiceRolePolicy`. Para obtener más información sobre la política, consulte AmazonOpenSearchServerlessServiceRolePolicy en la Guía de referencia de políticas administradas de AWS.

Permisos de rol vinculados a servicios para OpenSearch sin servidor

OpenSearch sin servidor utiliza el rol vinculado a un servicio denominado AWSServiceRoleForAmazonOpenSearchServerless, que permite que OpenSearch sin servidor llame a servicios de AWS en su nombre.

El rol vinculado a un servicio AWSServiceRoleForAmazonOpenSearchServerless confía en que los siguientes servicios asuman el rol:

- `observability.aoss.amazonaws.com`

La política de permisos del rol denominada `AmazonOpenSearchServerlessServiceRolePolicy` permite que OpenSearch sin servidor realice las siguientes acciones en los recursos especificados:

- Acción: `cloudwatch:PutMetricData` en todos los recursos de AWS.

> **Note**
>
> La política incluye la clave de condición `{"StringEquals": {"cloudwatch:namespace": "AWS/AOSS"}}`, lo que significa que el rol vinculado al servicio solo puede enviar datos métricos al espacio de nombres de `AWS/AOSS` CloudWatch.

Debe configurar permisos para permitir a una entidad de IAM (como un usuario, grupo o rol) crear, editar o eliminar un rol vinculado a servicios. Para obtener más información, consulte Permisos de roles vinculados a servicios en la Guía del usuario de IAM.

Creación del rol vinculado al servicio para OpenSearch sin servidor

No necesita crear manualmente un rol vinculado a servicios. Cuando se crea una colección de OpenSearch sin servidor en la Consola de administración de AWS, la AWS CLI, o la API de AWS, OpenSearch sin servidor crea el rol vinculado al servicio por usted.

> **Note**
>
> La primera vez que cree una colección, se le debe asignar la `iam:CreateServiceLinkedRole` en una política basada en identidades.

Si elimina este rol vinculado a servicios y necesita crearlo de nuevo, puede utilizar el mismo proceso para volver a crear el rol en su cuenta. Al crear una colección de OpenSearch sin servidor, OpenSearch sin servidor vuelve a crear el rol vinculado al servicio.

Además, puede utilizar la consola de IAM para crear un rol vinculado al servicio con el caso de uso de Amazon OpenSearch sin servidor. En la AWS CLI o AWS API, cree un rol vinculado al servicio con el nombre de servicio `observability.aoss.amazonaws.com`.

```
aws iam create-service-linked-role --aws-service-name
 "observability.aoss.amazonaws.com"
```

Para obtener más información, consulte Crear un rol vinculado a un servicio en la Guía del usuario de IAM. Si elimina este rol vinculado al servicio, puede utilizar este mismo proceso para volver a crear el rol.

Edición del rol vinculado a un servicio para OpenSearch sin servidor

OpenSearch sin servidor no permite editar el rol vinculado al servicio AWSServiceRoleForAmazonOpenSearchServerless. Después de crear un rol vinculado a un servicio, no puede cambiarle el nombre, ya que varias entidades pueden hacer referencia a él. Sin embargo, puede editar la descripción del rol mediante IAM. Para obtener más información, consulte Editar un rol vinculado a servicios en la Guía del usuario de IAM.

Eliminación del rol vinculado a un servicio para OpenSearch sin servidor

Si ya no necesita usar una característica o servicio que requieran un rol vinculado a un servicio, le recomendamos que elimine dicho rol. Esto evita tener una entidad sin uso que no se supervisa ni

mantiene activamente. Sin embargo, debe limpiar los recursos de su rol vinculado al servicio antes de eliminarlo manualmente.

Para eliminar AWSServiceRoleForAmazonOpenSearchServerless, primero, debe eliminar todas las colecciones de OpenSearch sin servidor en su Cuenta de AWS.

> **Note**
>
> Si OpenSearch sin servidor está utilizando el rol cuando intenta eliminar los recursos, entonces la eliminación podría fallar. En tal caso, espere unos minutos e intente de nuevo la operación.

Para eliminar manualmente el rol vinculado a servicios mediante IAM

Utilice la consola de IAM, la AWS CLI o la API de AWS para eliminar el rol vinculado al servicio AWSServiceRoleForAmazonOpenSearchServerless. Para obtener más información, consulte Eliminación de un rol vinculado a servicios en la Guía del usuario de IAM.

Regiones compatibles para roles vinculados a servicios de OpenSearch sin servidor

OpenSearch sin servidor admite el uso del rol vinculado al servicio AWSServiceRoleForAmazonOpenSearchServerless en todas las regiones en las que OpenSearch sin servidor esté disponible. Para obtener una lista de las regiones compatibles, consulte Puntos de conexión y cuotas de Amazon OpenSearch sin servidor en Referencia general de AWS.

Uso de roles vinculados a servicio para crear canalizaciones de OpenSearch Ingestion

Amazon OpenSearch Ingestion utiliza roles vinculados al servicio de AWS Identity and Access Management (IAM). Un rol vinculado a servicios es un tipo único de rol de IAM que está vinculado directamente a OpenSearch Ingestion. OpenSearch Ingestion predefine los roles vinculados a servicios y estos incluyen todos los permisos que el servicio requiere para llamar a otros servicios de AWS en su nombre.

OpenSearch Ingestion usa el rol vinculado al servicio AWSServiceRoleForAmazonOpenSearchIngestionService, excepto cuando usa una VPC autogestionada; en este caso, usa el rol vinculado al servicio

AWSServiceRoleForOpensearchIngestionSelfManagedVpce. La política adjunta proporciona los permisos necesarios para que el rol cree una nube privada virtual (VPC) entre su cuenta y OpenSearch Ingestion, y para publicar métricas de CloudWatch en su cuenta.

Permisos

El rol vinculado a servicios `AWSServiceRoleForAmazonOpenSearchIngestionService` confía en los siguientes servicios para asumir el rol:

- `osis.amazon.com`

La política de permisos del rol denominada `AmazonOpenSearchIngestionServiceRolePolicy` permite que OpenSearch Ingestion realice las siguientes acciones en los recursos especificados:

- Acción: `cloudwatch:PutMetricData` en `cloudwatch:namespace": "AWS/OSIS"`
- Acción: `ec2:CreateTags` en `arn:aws:ec2:*:*:network-interface/*`
- Acción: `ec2:CreateVpcEndpoint` en `*`
- Acción: `ec2:DeleteVpcEndpoints` en `*`
- Acción: `ec2:DescribeSecurityGroups` en `*`
- Acción: `ec2:DescribeSubnets` en `*`
- Acción: `ec2:DescribeVpcEndpoints` en `*`
- Acción: `ec2:ModifyVpcEndpoint` en `*`

El rol vinculado a servicios `AWSServiceRoleForOpensearchIngestionSelfManagedVpce` confía en los siguientes servicios para asumir el rol:

- `self-managed-vpce.osis.amazon.com`

La política de permisos del rol denominada `OpenSearchIngestionSelfManagedVpcePolicy` permite que OpenSearch Ingestion realice las siguientes acciones en los recursos especificados:

- Acción: `ec2:DescribeSubnets` en `*`
- Acción: `ec2:DescribeSecurityGroups` en `*`
- Acción: `ec2:DescribeVpcEndpoints` en `*`
- Acción: `cloudwatch:PutMetricData` en `cloudwatch:namespace": "AWS/OSIS"`

Debe configurar permisos para permitir a una entidad de IAM (como un usuario, grupo o rol) crear, editar o eliminar un rol vinculado a servicios. Para obtener más información, consulte Permisos de roles vinculados a servicios en la Guía del usuario de IAM.

Creación del rol vinculado a servicio para OpenSearch Ingestion

No necesita crear manualmente un rol vinculado a servicios. Cuando crea una canalización de OpenSearch Ingestion en la Consola de administración de AWS, la AWS CLI, o la API de AWS, OpenSearch Ingestion crea el rol vinculado al servicio por usted.

Si elimina este rol vinculado a servicios y necesita crearlo de nuevo, puede utilizar el mismo proceso para volver a crear el rol en su cuenta. Al crear una canalización de OpenSearch Ingestion, OpenSearch Ingestion vuelve a crear el rol vinculado al servicio.

Edición del rol vinculado a servicio para OpenSearch Ingestion

OpenSearch Ingestion no le permite editar el rol vinculado a servicios de `AWSServiceRoleForAmazonOpenSearchIngestionService`. Después de crear un rol vinculado al servicio, no podrá cambiar el nombre del rol, ya que varias entidades podrían hacer referencia al rol. Sin embargo, sí puede editar la descripción del rol con IAM. Para obtener más información, consulte Editar un rol vinculado a servicios en la Guía del usuario de IAM.

Eliminación del rol vinculado a un servicio para OpenSearch Ingestion

Si ya no necesita usar una característica o servicio que requieran un rol vinculado a un servicio, le recomendamos que elimine dicho rol. De esta forma no tiene una entidad no utilizada que no se monitoree ni mantenga de forma activa. Sin embargo, debe limpiar los recursos de su rol vinculado al servicio antes de eliminarlo manualmente.

Saneamiento de un rol vinculado a servicios

Antes de que pueda utilizar IAM para eliminar un rol vinculado a servicios, primero debe eliminar los recursos que utiliza el rol.

> **Note**
>
> Si OpenSearch Ingestion está utilizando el rol cuando intente eliminar los recursos, entonces la eliminación podría fallar. En tal caso, espere unos minutos e intente de nuevo la operación.

Eliminación de los recursos de OpenSearch Ingestion utilizados por el rol **`AWSServiceRoleForAmazonOpenSearchIngestionService`** o **`AWSServiceRoleForOpensearchIngestionSelfManagedVpce`**

1. Vaya a la consola de Amazon OpenSearch Service y elija Ingestion.
2. Elimine todas las canalizaciones. Para obtener instrucciones, consulte the section called "Eliminación de canalizaciones".

Elimine el rol vinculado a servicio para OpenSearch Ingestion

Puede utilizar la consola de OpenSearch Ingestion para eliminar un rol vinculado a un servicio.

Para eliminar un rol vinculado a un servicio (consola)

1. Desplácese hasta la consola de IAM.
2. Elija Roles y busque el rol AWSServiceRoleForAmazonOpenSearchIngestionService o AWSServiceRoleForOpensearchIngestionSelfManagedVpce.
3. Seleccione el rol y elija Eliminar.

Código de muestra para Amazon OpenSearch Service

Este capítulo contiene un código de muestra común para trabajar con Amazon OpenSearch Service: la firma de solicitudes HTTP en una variedad de lenguajes de programación, la compresión de cuerpos de solicitudes HTTP y la utilización de SDK de AWS para crear dominios.

Temas

- Compatibilidad con clientes de Elasticsearch
- Compresión de solicitudes HTTP en Amazon OpenSearch Service
- Utilizar SDK de AWS para interactuar con Amazon OpenSearch Service

Compatibilidad con clientes de Elasticsearch

Las versiones más recientes de los clientes de Elasticsearch pueden incluir verificaciones de licencia o versión que rompen artificialmente la compatibilidad. En la tabla siguiente, se incluyen recomendaciones sobre las versiones de esos clientes que se utilizarán para obtener la mejor compatibilidad con OpenSearch Service.

> **Important**
>
> Estas versiones de cliente están desactualizadas y no se actualizan con las dependencias más recientes, incluida Log4j. Recomendamos encarecidamente utilizar las versiones de OpenSearch de los clientes siempre que sea posible.

Cliente	Versión recomendada
Cliente REST de bajo nivel de Java	7.13.4
Cliente REST de alto nivel de Java	7.13.4
Cliente de Elasticsearch de Python	7.13.4
Cliente de Elasticsearch de Ruby	7.13.3
Cliente de Elasticsearch de Node.js	7.13.0

Compresión de solicitudes HTTP en Amazon OpenSearch Service

Puede comprimir solicitudes y respuestas HTTP en dominios de Amazon OpenSearch Service mediante la compresión gzip. La compresión gzip puede ayudar a reducir el tamaño de los documentos y a reducir la utilización y la latencia de la banda ancha, lo que permite mejorar las velocidades de transferencia.

La compresión gzip es compatible con todos los dominios que ejecutan OpenSearch, Elasticsearch 6.0 o posterior. Algunos clientes de OpenSearch cuentan con soporte integrado para la compresión gzip y muchos lenguajes de programación disponen de bibliotecas que simplifican el proceso.

Habilitar la compresión gzip

No debe confundirse con configuraciones similares de OpenSearch, `http_compression.enabled` es específico de OpenSearch Service y habilita o deshabilita la compresión gzip en un dominio. Los dominios que ejecutan OpenSearch o Elasticsearch 7.x tienen la compresión gzip habilitada de forma predeterminada, mientras que los dominios que ejecutan Elasticsearch 6.x la tienen desactivada de forma predeterminada.

Para habilitar la compresión gzip, envíe la siguiente solicitud:

```
PUT _cluster/settings
{
  "persistent" : {
    "http_compression.enabled": true
  }
}
```

Solicitudes a `_cluster/settings` deben estar descomprimidas, por lo que es posible que necesite utilizar un cliente independiente o una solicitud HTTP estándar para actualizar la configuración del clúster.

Para confirmar que ha habilitado correctamente la compresión gzip, envíe la siguiente solicitud:

```
GET _cluster/settings?include_defaults=true
```

Asegúrese de ver la siguiente configuración en la respuesta:

```
...
```

```
"http_compression": {
  "enabled": "true"
}
...
```

Encabezados obligatorios

Al incluir un cuerpo de la solicitud comprimido por gzip, mantenga el encabezado estándar `Content-Type: application/json` y agregue el encabezado `Content-Encoding: gzip`. Para aceptar una respuesta comprimida con gzip, agregue también el encabezado `Accept-Encoding: gzip`. Si un cliente de OpenSearch admite la compresión gzip, es probable que incluya estos encabezados automáticamente.

Código de muestra (Python 3)

En el siguiente ejemplo se utiliza opensearch-py para realizar la compresión y enviar la solicitud. Este código firma la solicitud mediante sus credenciales de IAM.

```
from opensearchpy import OpenSearch, RequestsHttpConnection
from requests_aws4auth import AWS4Auth
import boto3

host = '' # e.g. my-test-domain.us-east-1.es.amazonaws.com
region = '' # e.g. us-west-1
service = 'es'
credentials = boto3.Session().get_credentials()
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key, region, service,
 session_token=credentials.token)

# Create the client.
search = OpenSearch(
    hosts = [{'host': host, 'port': 443}],
    http_auth = awsauth,
    use_ssl = True,
    verify_certs = True,
    http_compress = True, # enables gzip compression for request bodies
    connection_class = RequestsHttpConnection
)

document = {
  "title": "Moneyball",
```

```
  "director": "Bennett Miller",
  "year": "2011"
}

# Send the request.
print(search.index(index='movies', id='1', body=document, refresh=True))

# print(search.index(index='movies', doc_type='_doc', id='1', body=document,
 refresh=True))
```

Como alternativa, puede especificar los encabezados adecuados, comprimir el cuerpo de la solicitud usted mismo y utilizar una biblioteca HTTP estándar como Solicitudes. Este código firma la solicitud con credenciales básicas HTTP, que podrían ser compatibles con el dominio si utiliza el control de acceso detallado.

```
import requests
import gzip
import json

base_url = '' # The domain with https:// and a trailing slash. For example, https://my-
test-domain.us-east-1.es.amazonaws.com/
auth = ('master-user', 'master-user-password') # For testing only. Don't store
 credentials in code.

headers = {'Accept-Encoding': 'gzip', 'Content-Type': 'application/json',
           'Content-Encoding': 'gzip'}

document = {
  "title": "Moneyball",
  "director": "Bennett Miller",
  "year": "2011"
}

# Compress the document.
compressed_document = gzip.compress(json.dumps(document).encode())

# Send the request.
path = 'movies/_doc?refresh=true'
url = base_url + path
response = requests.post(url, auth=auth, headers=headers, data=compressed_document)
print(response.status_code)
print(response.text)
```

Utilizar SDK de AWS para interactuar con Amazon OpenSearch Service

En esta sección, se incluyen ejemplos de cómo utilizar los SDK de AWS para interactuar con la API de configuración de Amazon OpenSearch Service. Estos ejemplos de código muestran cómo crear, actualizar y eliminar dominios de OpenSearch Service.

Java

Esta sección incluye ejemplos para las versiones 1 y 2 del AWS SDK para Java.

Version 2

En este ejemplo se utiliza el constructor OpenSearchClientBuilder de la versión 2 de AWS SDK para Java para crear un dominio OpenSearch, actualizar su configuración y eliminarlo. Quite los comentarios de las llamadas a `waitForDomainProcessing` (y el comentario de la llamada a `deleteDomain`) para permitir que se active el dominio y que se pueda utilizar.

```
package com.example.samples;

import java.util.concurrent.TimeUnit;
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.opensearch.OpenSearchClient;
import software.amazon.awssdk.services.opensearch.model.ClusterConfig;
import software.amazon.awssdk.services.opensearch.model.EBSOptions;
import software.amazon.awssdk.services.opensearch.model.CognitoOptions;
import software.amazon.awssdk.services.opensearch.model.NodeToNodeEncryptionOptions;
import software.amazon.awssdk.services.opensearch.model.CreateDomainRequest;
import software.amazon.awssdk.services.opensearch.model.CreateDomainResponse;
import software.amazon.awssdk.services.opensearch.model.DescribeDomainRequest;
import software.amazon.awssdk.services.opensearch.model.UpdateDomainConfigRequest;
import software.amazon.awssdk.services.opensearch.model.UpdateDomainConfigResponse;
import software.amazon.awssdk.services.opensearch.model.DescribeDomainResponse;
import software.amazon.awssdk.services.opensearch.model.DeleteDomainRequest;
import software.amazon.awssdk.services.opensearch.model.DeleteDomainResponse;
import software.amazon.awssdk.services.opensearch.model.OpenSearchException;
import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;

/**
 * Sample class demonstrating how to use the Amazon Web Services SDK for Java to
 create, update,
```

```
 * and delete Amazon OpenSearch Service domains.
 */

public class OpenSearchSample {

    public static void main(String[] args) {

     String domainName = "my-test-domain";

     // Build the client using the default credentials chain.
        // You can use the CLI and run `aws configure` to set access key, secret
        // key, and default region.

        OpenSearchClient client = OpenSearchClient.builder()
          // Unnecessary, but lets you use a region different than your default.
          .region(Region.US_EAST_1)
          // Unnecessary, but if desired, you can use a different provider chain.
          .credentialsProvider(DefaultCredentialsProvider.create())
                     .build();

        // Create a new domain, update its configuration, and delete it.
        createDomain(client, domainName);
        //waitForDomainProcessing(client, domainName);
        updateDomain(client, domainName);
        //waitForDomainProcessing(client, domainName);
        deleteDomain(client, domainName);
    }

    /**
     * Creates an Amazon OpenSearch Service domain with the specified options.
     * Some options require other Amazon Web Services resources, such as an Amazon
 Cognito user pool
     * and identity pool, whereas others require just an instance type or instance
     * count.
     *
     * @param client
     *            The client to use for the requests to Amazon OpenSearch Service
     * @param domainName
     *            The name of the domain you want to create
     */

    public static void createDomain(OpenSearchClient client, String domainName) {

     // Create the request and set the desired configuration options
```

```
        try {

            ClusterConfig clusterConfig = ClusterConfig.builder()
                    .dedicatedMasterEnabled(true)
                    .dedicatedMasterCount(3)
                    // Small, inexpensive instance types for testing. Not
 recommended for production.
                    .dedicatedMasterType("t2.small.search")
                    .instanceType("t2.small.search")
                    .instanceCount(5)
                    .build();

            // Many instance types require EBS storage.
            EBSOptions ebsOptions = EBSOptions.builder()
                    .ebsEnabled(true)
                    .volumeSize(10)
                    .volumeType("gp2")
                    .build();

            NodeToNodeEncryptionOptions encryptionOptions =
 NodeToNodeEncryptionOptions.builder()
                    .enabled(true)
                    .build();

            CreateDomainRequest createRequest = CreateDomainRequest.builder()
                    .domainName(domainName)
                    .engineVersion("OpenSearch_1.0")
                    .clusterConfig(clusterConfig)
                    .ebsOptions(ebsOptions)
                    .nodeToNodeEncryptionOptions(encryptionOptions)
                    // You can uncomment this line and add your account ID, a
 username, and the
                    // domain name to add an access policy.
                    // .accessPolicies("{\"Version\":\"2012-10-17\",
\"Statement\":[{\"Effect\":\"Allow\",\"Principal\":{\"AWS\":
[\"arn:aws:iam::123456789012:user/user-name\"]},\"Action\":[\"es:*\"],\"Resource\":
\"arn:aws:es:region:123456789012:domain/domain-name/*\"}]}")
                    .build();

            // Make the request.
            System.out.println("Sending domain creation request...");
            CreateDomainResponse createResponse =
 client.createDomain(createRequest);
```

```
            System.out.println("Domain status:
"+createResponse.domainStatus().toString());
            System.out.println("Domain ID:
"+createResponse.domainStatus().domainId());


        } catch (OpenSearchException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
    }

    /**
     * Updates the configuration of an Amazon OpenSearch Service domain with the
     * specified options. Some options require other Amazon Web Services resources,
such as an
     * Amazon Cognito user pool and identity pool, whereas others require just an
     * instance type or instance count.
     *
     * @param client
     *             The client to use for the requests to Amazon OpenSearch Service
     * @param domainName
     *             The name of the domain to update
     */

    public static void updateDomain(OpenSearchClient client, String domainName) {

     // Updates the domain to use three data instances instead of five.
        // You can uncomment the Cognito line and fill in the strings to enable
Cognito
        // authentication for OpenSearch Dashboards.

        try {

            ClusterConfig clusterConfig = ClusterConfig.builder()
                    .instanceCount(5)
                    .build();

            CognitoOptions cognitoOptions = CognitoOptions.builder()
                    .enabled(true)
                    .userPoolId("user-pool-id")
                    .identityPoolId("identity-pool-id")
                    .roleArn("role-arn")
                    .build();
```

```
            UpdateDomainConfigRequest updateRequest =
UpdateDomainConfigRequest.builder()
                    .domainName(domainName)
                    .clusterConfig(clusterConfig)
                    //.cognitoOptions(cognitoOptions)
                    .build();

            System.out.println("Sending domain update request...");
            UpdateDomainConfigResponse updateResponse =
client.updateDomainConfig(updateRequest);
            System.out.println("Domain config:
"+updateResponse.domainConfig().toString());


        } catch (OpenSearchException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
    }

    /**
     * Deletes an Amazon OpenSearch Service domain. Deleting a domain can take
     * several minutes.
     *
     * @param client
     *             The client to use for the requests to Amazon OpenSearch Service
     * @param domainName
     *             The name of the domain that you want to delete
     */

    public static void deleteDomain(OpenSearchClient client, String domainName) {

        try {

            DeleteDomainRequest deleteRequest = DeleteDomainRequest.builder()
                    .domainName(domainName)
                    .build();

            System.out.println("Sending domain deletion request...");
            DeleteDomainResponse deleteResponse =
client.deleteDomain(deleteRequest);
            System.out.println("Domain status: "+deleteResponse.toString());
```

```
        } catch (OpenSearchException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
    }

    /**
     * Waits for the domain to finish processing changes. New domains typically take
15-30 minutes
     * to initialize, but can take longer depending on the configuration. Most
updates to existing domains
     * take a similar amount of time. This method checks every 15 seconds and
finishes only when
     * the domain's processing status changes to false.
     *
     * @param client
     *            The client to use for the requests to Amazon OpenSearch Service
     * @param domainName
     *            The name of the domain that you want to check
     */

    public static void waitForDomainProcessing(OpenSearchClient client, String
domainName) {
        // Create a new request to check the domain status.
        DescribeDomainRequest describeRequest = DescribeDomainRequest.builder()
                .domainName(domainName)
                .build();

        // Every 15 seconds, check whether the domain is processing.
        DescribeDomainResponse describeResponse =
client.describeDomain(describeRequest);
        while (describeResponse.domainStatus().processing()) {
            try {
                System.out.println("Domain still processing...");
                TimeUnit.SECONDS.sleep(15);
                describeResponse = client.describeDomain(describeRequest);
            } catch (InterruptedException e) {
                e.printStackTrace();
            }
        }

        // Once we exit that loop, the domain is available
```

```
        System.out.println("Amazon OpenSearch Service has finished processing
 changes for your domain.");
        System.out.println("Domain description: "+describeResponse.toString());
    }
}
```

Version 1

En este ejemplo se utiliza el constructor AWSElasticsearchClientBuilder de la versión 1 del AWS SDK para Java para crear un dominio Elasticsearch heredado, actualizar su configuración y eliminarlo. Quite los comentarios de las llamadas a `waitForDomainProcessing` (y el comentario de la llamada a `deleteDomain`) para permitir que se active el dominio y que se pueda utilizar.

```
package com.amazonaws.samples;

import java.util.concurrent.TimeUnit;
import com.amazonaws.auth.DefaultAWSCredentialsProviderChain;
import com.amazonaws.regions.Regions;
import com.amazonaws.services.elasticsearch.AWSElasticsearch;
import com.amazonaws.services.elasticsearch.AWSElasticsearchClientBuilder;
import com.amazonaws.services.elasticsearch.model.CreateElasticsearchDomainRequest;
import com.amazonaws.services.elasticsearch.model.CreateElasticsearchDomainResult;
import com.amazonaws.services.elasticsearch.model.DeleteElasticsearchDomainRequest;
import com.amazonaws.services.elasticsearch.model.DeleteElasticsearchDomainResult;
import
 com.amazonaws.services.elasticsearch.model.DescribeElasticsearchDomainRequest;
import com.amazonaws.services.elasticsearch.model.DescribeElasticsearchDomainResult;
import com.amazonaws.services.elasticsearch.model.EBSOptions;
import com.amazonaws.services.elasticsearch.model.ElasticsearchClusterConfig;
import com.amazonaws.services.elasticsearch.model.ResourceNotFoundException;
import
 com.amazonaws.services.elasticsearch.model.UpdateElasticsearchDomainConfigRequest;
import
 com.amazonaws.services.elasticsearch.model.UpdateElasticsearchDomainConfigResult;
import com.amazonaws.services.elasticsearch.model.VolumeType;

/**
 * Sample class demonstrating how to use the Amazon Web Services SDK for Java to
 create, update,
 * and delete Amazon OpenSearch Service domains.
 */
```

```
public class OpenSearchSample {

    public static void main(String[] args) {

        final String domainName = "my-test-domain";

        // Build the client using the default credentials chain.
        // You can use the CLI and run `aws configure` to set access key, secret
        // key, and default region.
        final AWSElasticsearch client = AWSElasticsearchClientBuilder
                .standard()
                // Unnecessary, but lets you use a region different than your
 default.
                .withRegion(Regions.US_WEST_2)
                // Unnecessary, but if desired, you can use a different provider
 chain.
                .withCredentials(new DefaultAWSCredentialsProviderChain())
                .build();

        // Create a new domain, update its configuration, and delete it.
        createDomain(client, domainName);
        // waitForDomainProcessing(client, domainName);
        updateDomain(client, domainName);
        // waitForDomainProcessing(client, domainName);
        deleteDomain(client, domainName);
    }

    /**
     * Creates an Amazon OpenSearch Service domain with the specified options.
     * Some options require other Amazon Web Services resources, such as an Amazon
 Cognito user pool
     * and identity pool, whereas others require just an instance type or instance
     * count.
     *
     * @param client
     *            The client to use for the requests to Amazon OpenSearch Service
     * @param domainName
     *            The name of the domain you want to create
     */
    private static void createDomain(final AWSElasticsearch client, final String
 domainName) {

        // Create the request and set the desired configuration options
```

```
        CreateElasticsearchDomainRequest createRequest = new
 CreateElasticsearchDomainRequest()
                .withDomainName(domainName)
                .withElasticsearchVersion("7.10")
                .withElasticsearchClusterConfig(new ElasticsearchClusterConfig()
                        .withDedicatedMasterEnabled(true)
                        .withDedicatedMasterCount(3)
                        // Small, inexpensive instance types for testing. Not
 recommended for production
                        // domains.
                        .withDedicatedMasterType("t2.small.elasticsearch")
                        .withInstanceType("t2.small.elasticsearch")
                        .withInstanceCount(5))
                // Many instance types require EBS storage.
                .withEBSOptions(new EBSOptions()
                        .withEBSEnabled(true)
                        .withVolumeSize(10)
                        .withVolumeType(VolumeType.Gp2));
                // You can uncomment this line and add your account ID, a username,
 and the
                // domain name to add an access policy.
                // .withAccessPolicies("{\"Version\":\"2012-10-17\",
\"Statement\":[{\"Effect\":\"Allow\",\"Principal\":{\"AWS\":
[\"arn:aws:iam::123456789012:user/user-name\"]},\"Action\":[\"es:*\"],\"Resource\":
\"arn:aws:es:region:123456789012:domain/domain-name/*\"}]}")


        // Make the request.
        System.out.println("Sending domain creation request...");
        CreateElasticsearchDomainResult createResponse =
 client.createElasticsearchDomain(createRequest);
        System.out.println("Domain creation response from Amazon OpenSearch
 Service:");
        System.out.println(createResponse.getDomainStatus().toString());
    }

    /**
     * Updates the configuration of an Amazon OpenSearch Service domain with the
     * specified options. Some options require other Amazon Web Services resources,
 such as an
     * Amazon Cognito user pool and identity pool, whereas others require just an
     * instance type or instance count.
     *
     * @param client
```

```
     *            The client to use for the requests to Amazon OpenSearch Service
     * @param domainName
     *            The name of the domain to update
     */
    private static void updateDomain(final AWSElasticsearch client, final String
domainName) {
        try {
            // Updates the domain to use three data instances instead of five.
            // You can uncomment the Cognito lines and fill in the strings to enable
Cognito
            // authentication for OpenSearch Dashboards.
            final UpdateElasticsearchDomainConfigRequest updateRequest = new
UpdateElasticsearchDomainConfigRequest()
                    .withDomainName(domainName)
                    // .withCognitoOptions(new CognitoOptions()
                            // .withEnabled(true)
                            // .withUserPoolId("user-pool-id")
                            // .withIdentityPoolId("identity-pool-id")
                            // .withRoleArn("role-arn")
                    .withElasticsearchClusterConfig(new ElasticsearchClusterConfig()
                            .withInstanceCount(3));

            System.out.println("Sending domain update request...");
            final UpdateElasticsearchDomainConfigResult updateResponse = client
                    .updateElasticsearchDomainConfig(updateRequest);
            System.out.println("Domain update response from Amazon OpenSearch
Service:");
            System.out.println(updateResponse.toString());
        } catch (ResourceNotFoundException e) {
            System.out.println("Domain not found. Please check the domain name.");
        }
    }

    /**
     * Deletes an Amazon OpenSearch Service domain. Deleting a domain can take
     * several minutes.
     *
     * @param client
     *            The client to use for the requests to Amazon OpenSearch Service
     * @param domainName
     *            The name of the domain that you want to delete
     */
    private static void deleteDomain(final AWSElasticsearch client, final String
domainName) {
```

```
        try {
            final DeleteElasticsearchDomainRequest deleteRequest = new
DeleteElasticsearchDomainRequest()
                    .withDomainName(domainName);

            System.out.println("Sending domain deletion request...");
            final DeleteElasticsearchDomainResult deleteResponse =
client.deleteElasticsearchDomain(deleteRequest);
            System.out.println("Domain deletion response from Amazon OpenSearch
Service:");
            System.out.println(deleteResponse.toString());
        } catch (ResourceNotFoundException e) {
            System.out.println("Domain not found. Please check the domain name.");
        }
    }

    /**
     * Waits for the domain to finish processing changes. New domains typically take
15-30 minutes
     * to initialize, but can take longer depending on the configuration. Most
updates to existing domains
     * take a similar amount of time. This method checks every 15 seconds and
finishes only when
     * the domain's processing status changes to false.
     *
     * @param client
     *            The client to use for the requests to Amazon OpenSearch Service
     * @param domainName
     *            The name of the domain that you want to check
     */
    private static void waitForDomainProcessing(final AWSElasticsearch client, final
String domainName) {
        // Create a new request to check the domain status.
        final DescribeElasticsearchDomainRequest describeRequest = new
DescribeElasticsearchDomainRequest()
                .withDomainName(domainName);

        // Every 15 seconds, check whether the domain is processing.
        DescribeElasticsearchDomainResult describeResponse =
client.describeElasticsearchDomain(describeRequest);
        while (describeResponse.getDomainStatus().isProcessing()) {
            try {
                System.out.println("Domain still processing...");
                TimeUnit.SECONDS.sleep(15);
```

```
                describeResponse =
 client.describeElasticsearchDomain(describeRequest);
            } catch (InterruptedException e) {
                e.printStackTrace();
            }
        }

        // Once we exit that loop, the domain is available
        System.out.println("Amazon OpenSearch Service has finished processing
 changes for your domain.");
        System.out.println("Domain description response from Amazon OpenSearch
 Service:");
        System.out.println(describeResponse.toString());
    }
}
```

Python

En este ejemplo se utiliza el cliente Python de bajo nivel OpenSearchService desde AWS SDK for Python (Boto) para crear un dominio, actualizar su configuración y eliminarlo.

```
import boto3
import botocore
from botocore.config import Config
import time

# Build the client using the default credential configuration.
# You can use the CLI and run 'aws configure' to set access key, secret
# key, and default region.

my_config = Config(
    # Optionally lets you specify a region other than your default.
    region_name='us-west-2'
)

client = boto3.client('opensearch', config=my_config)

domainName = 'my-test-domain'  # The name of the domain


def createDomain(client, domainName):
    """Creates an Amazon OpenSearch Service domain with the specified options."""
```

```
    response = client.create_domain(
        DomainName=domainName,
        EngineVersion='OpenSearch_1.0',
        ClusterConfig={
            'InstanceType': 't2.small.search',
            'InstanceCount': 5,
            'DedicatedMasterEnabled': True,
            'DedicatedMasterType': 't2.small.search',
            'DedicatedMasterCount': 3
        },
        # Many instance types require EBS storage.
        EBSOptions={
            'EBSEnabled': True,
            'VolumeType': 'gp2',
            'VolumeSize': 10
        },
        AccessPolicies="{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":\"Allow
\",\"Principal\":{\"AWS\":[\"arn:aws:iam::123456789012:user/user-name\"]},\"Action\":
[\"es:*\"],\"Resource\":\"arn:aws:es:us-west-2:123456789012:domain/my-test-domain/*
\"}]}",
        NodeToNodeEncryptionOptions={
            'Enabled': True
        }
    )
    print("Creating domain...")
    print(response)


def updateDomain(client, domainName):
    """Updates the domain to use three data nodes instead of five."""
    try:
        response = client.update_domain_config(
            DomainName=domainName,
            ClusterConfig={
                'InstanceCount': 3
            }
        )
        print('Sending domain update request...')
        print(response)

    except botocore.exceptions.ClientError as error:
        if error.response['Error']['Code'] == 'ResourceNotFoundException':
            print('Domain not found. Please check the domain name.')
        else:
```

```
            raise error


def deleteDomain(client, domainName):
    """Deletes an OpenSearch Service domain. Deleting a domain can take several
 minutes."""
    try:
        response = client.delete_domain(
            DomainName=domainName
        )
        print('Sending domain deletion request...')
        print(response)

    except botocore.exceptions.ClientError as error:
        if error.response['Error']['Code'] == 'ResourceNotFoundException':
            print('Domain not found. Please check the domain name.')
        else:
            raise error


def waitForDomainProcessing(client, domainName):
    """Waits for the domain to finish processing changes."""
    try:
        response = client.describe_domain(
            DomainName=domainName
        )
        # Every 15 seconds, check whether the domain is processing.
        while response["DomainStatus"]["Processing"] == True:
            print('Domain still processing...')
            time.sleep(15)
            response = client.describe_domain(
                DomainName=domainName)

        # Once we exit the loop, the domain is available.
        print('Amazon OpenSearch Service has finished processing changes for your
 domain.')
        print('Domain description:')
        print(response)

    except botocore.exceptions.ClientError as error:
        if error.response['Error']['Code'] == 'ResourceNotFoundException':
            print('Domain not found. Please check the domain name.')
        else:
            raise error
```

```
def main():
    """Create a new domain, update its configuration, and delete it."""
    createDomain(client, domainName)
    waitForDomainProcessing(client, domainName)
    updateDomain(client, domainName)
    waitForDomainProcessing(client, domainName)
    deleteDomain(client, domainName)
```

Nodo

En este ejemplo, se utiliza la versión 3 del SDK para JavaScript in Node.js OpenSearch client para crear un dominio, actualizar su configuración y eliminarlo.

```
var {
    OpenSearchClient,
    CreateDomainCommand,
    DescribeDomainCommand,
    UpdateDomainConfigCommand,
    DeleteDomainCommand
} = require("@aws-sdk/client-opensearch");
var sleep = require('sleep');

var client = new OpenSearchClient();

var domainName = 'my-test-domain'

// Create a new domain, update its configuration, and delete it.
createDomain(client, domainName)
waitForDomainProcessing(client, domainName)
updateDomain(client, domainName)
waitForDomainProcessing(client, domainName)
deleteDomain(client, domainName)

async function createDomain(client, domainName) {
    // Creates an Amazon OpenSearch Service domain with the specified options.
    var command = new CreateDomainCommand({
        DomainName: domainName,
        EngineVersion: 'OpenSearch_1.0',
        ClusterConfig: {
        'InstanceType': 't2.small.search',
        'InstanceCount': 5,
```

```
        'DedicatedMasterEnabled': 'True',
        'DedicatedMasterType': 't2.small.search',
        'DedicatedMasterCount': 3
        },
        EBSOptions:{
            'EBSEnabled': 'True',
            'VolumeType': 'gp2',
            'VolumeSize': 10
        },
        AccessPolicies: "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":\"Allow
\",\"Principal\":{\"AWS\":[\"arn:aws:iam::123456789012:user/user-name\"]},\"Action\":
[\"es:*\"],\"Resource\":\"arn:aws:es:us-east-1:123456789012:domain/my-test-domain/*
\"}]}",
         NodeToNodeEncryptionOptions:{
            'Enabled': 'True'
        }
    });
    const response = await client.send(command);
    console.log("Creating domain...");
    console.log(response);
}

async function updateDomain(client, domainName) {
    // Updates the domain to use three data nodes instead of five.
    var command = new UpdateDomainConfigCommand({
        DomainName: domainName,
        ClusterConfig: {
        'InstanceCount': 3
        }
    });
    const response = await client.send(command);
    console.log('Sending domain update request...');
    console.log(response);
}

async function deleteDomain(client, domainName) {
    // Deletes an OpenSearch Service domain. Deleting a domain can take several
 minutes.
    var command = new DeleteDomainCommand({
        DomainName: domainName
    });
    const response = await client.send(command);
    console.log('Sending domain deletion request...');
    console.log(response);
```

```
}

async function waitForDomainProcessing(client, domainName) {
    // Waits for the domain to finish processing changes.
    try {
        var command = new DescribeDomainCommand({
            DomainName: domainName
        });
        var response = await client.send(command);

        while (response.DomainStatus.Processing == true) {
            console.log('Domain still processing...')
            await sleep(15000) // Wait for 15 seconds, then check the status again
            function sleep(ms) {
                return new Promise((resolve) => {
                    setTimeout(resolve, ms);
                });
            }
            var response = await client.send(command);
        }
        // Once we exit the loop, the domain is available.
        console.log('Amazon OpenSearch Service has finished processing changes for your
 domain.');
        console.log('Domain description:');
        console.log(response);

    } catch (error) {
        if (error.name === 'ResourceNotFoundException') {
            console.log('Domain not found. Please check the domain name.');
            }
    };
}
```

Indexación de datos en Amazon OpenSearch Service

Dado que Amazon OpenSearch Service utiliza una API de REST, existen numerosos métodos para indexar documentos. Puede utilizar clientes estándar como curl o cualquier lenguaje de programación que pueda enviar solicitudes HTTP. Para simplificar aún más el proceso de interacción, OpenSearch Service cuenta con clientes para diversos lenguajes de programación. Los usuarios avanzados pueden pasar directamente a the section called “Carga de datos de streaming en OpenSearch Service”.

Le recomendamos mucho que utilice Amazon OpenSearch Ingestion para incorporar datos, que es un recopilador de datos totalmente administrado e integrado en OpenSearch Service. Para más información, consulte Amazon OpenSearch Ingestion.

Para ver una introducción a la indexación, consulte la documentación de OpenSearch.

Restricciones de nomenclatura de los índices

Los índices de OpenSearch Service tienen las siguientes restricciones de nomenclatura:

- Todas las letras deben estar en minúsculas.
- Los nombres de los índices no pueden empezar por `_` ni `-`.
- Los nombres de los índices no pueden contener espacios, comas, `:`, `"`, `*`, `+`, `/`, `\`, `|`, `?`, `#`, `>` o `<`.

No incluya información confidencial en los nombres de ID de índice, tipo o documento. OpenSearch Service utiliza estos nombres en sus identificadores uniformes de recursos (URI). A menudo, los servidores y las aplicaciones registran solicitudes HTTP, lo que puede dar lugar a que se expongan datos innecesariamente si los URI contienen información confidencial:

```
2018-10-03T23:39:43 198.51.100.14 200 "GET https://opensearch-domain/dr-jane-doe/flu-patients-2018/202-555-0100/ HTTP/1.1"
```

Aunque no tenga permisos para ver el documento JSON asociado, puede inferir de esta línea de registro falsa que uno de los pacientes del Dr. Doe, cuyo número de teléfono es el 202-555-0100, tuvo gripe en 2018.

Si OpenSearch Service detecta una dirección IP real o percibida en un nombre de índice (por ejemplo, `my-index-12.34.56.78.91`), enmascara la dirección IP. Una llamada a `_cat/indices` produce la siguiente respuesta:

```
green open my-index-x.x.x.x.91    soY19tBERoKo71WcEScidw 5 1 0 0   2kb  1kb
```

Para evitar confusiones innecesarias, evite incluir direcciones IP en los nombres de los índices.

Reducción del tamaño de la respuesta

Las respuestas de las API `_index` y `_bulk` contienen bastante información. Esta información puede resultar útil para resolver problemas de las solicitudes o para implementar la lógica de reintentos, aunque puede utilizar una banda ancha considerable. En este ejemplo, la indexación de un documento de 32 bytes se traduce en una respuesta de 339 bytes (incluidos los encabezados):

```
PUT opensearch-domain/more-movies/_doc/1
{"title": "Back to the Future"}
```

Respuesta

```
{
  "_index": "more-movies",
  "_type": "_doc",
  "_id": "1",
  "_version": 4,
  "result": "updated",
  "_shards": {
    "total": 2,
    "successful": 2,
    "failed": 0
  },
  "_seq_no": 3,
  "_primary_term": 1
}
```

Este tamaño de respuesta puede parecer mínimo, pero si indexa 1 000 000 de documentos al día (aproximadamente 11,5 documentos por segundo), 339 bytes por respuesta equivalen a 10,17 GB de tráfico de descarga al mes.

Si los costos de transferencia de datos son una preocupación, utilice el parámetro `filter_path` para reducir el tamaño de la respuesta de OpenSearch Service, pero procure no filtrar campos que necesita para poder identificar o reintentar solicitudes que fallaron. Estos campos varían por cliente. El parámetro `filter_path` funciona para todas las API de REST de OpenSearch Service, pero resulta especialmente útil con las API a las que se llama con frecuencia, como las API `_index` y `_bulk`:

```
PUT opensearch-domain/more-movies/_doc/1?filter_path=result,_shards.total
{"title": "Back to the Future"}
```

Respuesta

```
{
  "result": "updated",
  "_shards": {
    "total": 2
  }
}
```

En lugar de incluir campos, puede excluir campos con un prefijo `-`. `filter_path` también es compatible con comodines:

```
POST opensearch-domain/_bulk?filter_path=-took,-items.index._*
{ "index": { "_index": "more-movies", "_id": "1" } }
{"title": "Back to the Future"}
{ "index": { "_index": "more-movies", "_id": "2" } }
{"title": "Spirited Away"}
```

Respuesta

```
{
  "errors": false,
  "items": [
    {
      "index": {
        "result": "updated",
        "status": 200
      }
    },
    {
```

```
      "index": {
        "result": "updated",
        "status": 200
      }
    }
  ]
}
```

Códecs de índice

Los códecs de índice determinan cómo se comprimen y almacenan en el disco los campos almacenados en un índice. El códec de índice se controla mediante la configuración `index.codec` estática, que especifica el algoritmo de compresión. Esta configuración afecta el tamaño de la partición del índice y el rendimiento de la operación.

Para obtener una lista de los códecs compatibles y sus características de rendimiento, consulte los Códecs compatibles en la documentación de OpenSearch.

Cuando elija un códec de índice, tenga en cuenta lo siguiente:

- Para evitar los problemas que supone cambiar la configuración del códec de un índice existente, pruebe una carga de trabajo representativa en un entorno que no sea de producción antes de utilizar una nueva configuración de códec. Para más información, consulte Cambio de un códec de índice.
- No puede usar los códecs de compresión Zstandard (`"index.codec": "zstd"` o `"index.codec": "zstd_no_dict"`) para los índices k-NN o Security Analytics.

Carga de datos de streaming en Amazon OpenSearch Service

Puede usar OpenSearch Ingestion para cargar directamente los datos de streaming en su dominio de Amazon OpenSearch Service, sin necesidad de utilizar soluciones de terceros. Para enviar datos a OpenSearch Ingestion, debe configurar los productores de datos y el servicio envía automáticamente los datos al dominio o a la colección que especifique. Para comenzar con OpenSearch Ingestion, consulte the section called “Tutorial: incorporar datos en una colección”.

Puede seguir utilizando otras fuentes para cargar datos de streaming, como Amazon Data Firehose y Registros de Amazon CloudWatch, que cuentan con compatibilidad integrada con OpenSearch Service. Otros, como Amazon S3, Amazon Kinesis Data Streams y Amazon DynamoDB, utilizan

funciones AWS Lambda como controladores de eventos. Las funciones de Lambda responden a los nuevos datos procesándolos y transmitiéndolos al dominio.

> **Note**
>
> Lambda es compatible con varios lenguajes de programación populares y está disponible en la mayoría de las Regiones de AWS. Para más información, consulte Introducción a Lambda en la Guía para desarrolladores de AWS Lambda y Puntos de conexión de servicio de AWS en Referencia general de AWS.

Carga de datos de streaming desde OpenSearch Ingestion

Puede utilizar Amazon OpenSearch Ingestion para cargar datos en un dominio de OpenSearch Service. Puede configurar los productores de datos para que envíen datos a OpenSearch Ingestion, que a su vez los envían automáticamente a la colección que especifique. También puede configurar OpenSearch Ingestion para transformar los datos antes de entregarlos. Para más información, consulte Amazon OpenSearch Ingestion.

Carga de datos de streaming desde Amazon S3

Puede utilizar Lambda para enviar datos al dominio del servicio OpenSearch Service desde Amazon S3. Cuando llegan datos nuevos a un bucket de S3, activan una notificación de eventos en Lambda que, a su vez, ejecuta el código personalizado para realizar la indexación.

Este método de streaming de datos es extremadamente flexible. Es posible indexar los metadatos del objeto o, si el objeto contiene texto sin formato, analizar e indexar algunos elementos del cuerpo del objeto. En esta sección, se incluye código de muestra sencillo de Python que utiliza expresiones regulares para analizar un archivo de registros e indexar los resultados obtenidos.

Requisitos previos

Antes de continuar, debe contar con los siguientes recursos.

Requisito previo	Descripción
Bucket de Amazon S3	Para más información, consulte Crear su primer bucket de S3 en la Guía del usuario de Amazon Simple Storage Service. El bucket debe encontrarse en la misma región que el dominio de OpenSearch Service.

Requisito previo	Descripción
Dominio de OpenSearch Service	Es el destino de los datos después de que la función de Lambda los procesa. Para más información, consulte the section called “ Creación de dominios de OpenSearch Service”.

Crear el paquete de implementación de Lambda

Los paquetes de implementación son archivos ZIP o JAR que contienen el código y sus dependencias. En esta sección, se incluye un código de muestra de Python. Para otros lenguajes de programación, consulte Paquetes de implementación de Lambda en la Guía para desarrolladores de AWS Lambda.

1. Cree un directorio. En este ejemplo, utilizamos el nombre `s3-to-opensearch`.
2. Cree un archivo en el directorio denominado `sample.py`:

```
import boto3
import re
import requests
from requests_aws4auth import AWS4Auth

region = '' # e.g. us-west-1
service = 'es'
credentials = boto3.Session().get_credentials()
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key, region, service,
 session_token=credentials.token)

host = '' # the OpenSearch Service domain, e.g. https://search-mydomain.us-
west-1.es.amazonaws.com
index = 'lambda-s3-index'
datatype = '_doc'
url = host + '/' + index + '/' + datatype

headers = { "Content-Type": "application/json" }

s3 = boto3.client('s3')

# Regular expressions used to parse some simple log lines
ip_pattern = re.compile('(\d+\.\d+\.\d+\.\d+)')
time_pattern = re.compile('\[(\d+\/\w\w\w\/\d\d\d\d:\d\d:\d\d:\d\d\s-\d\d\d\d)\]')
```

```
message_pattern = re.compile('\"(.+)\"')

# Lambda execution starts here
def handler(event, context):
    for record in event['Records']:

        # Get the bucket name and key for the new file
        bucket = record['s3']['bucket']['name']
        key = record['s3']['object']['key']

        # Get, read, and split the file into lines
        obj = s3.get_object(Bucket=bucket, Key=key)
        body = obj['Body'].read()
        lines = body.splitlines()

        # Match the regular expressions to each line and index the JSON
        for line in lines:
            line = line.decode("utf-8")
            ip = ip_pattern.search(line).group(1)
            timestamp = time_pattern.search(line).group(1)
            message = message_pattern.search(line).group(1)

            document = { "ip": ip, "timestamp": timestamp, "message": message }
            r = requests.post(url, auth=awsauth, json=document, headers=headers)
```

Edite las variables para `region` y `host`.

3. Instale pip, si todavía no lo hizo, luego instale las dependencias en un directorio `package` nuevo:

```
cd s3-to-opensearch

pip install --target ./package requests
pip install --target ./package requests_aws4auth
```

Todos los entornos de ejecución de Lambda tienen instalado Boto3, por lo que no es necesario incluirlo en el paquete de implementación.

4. Cree el paquete con el código de la aplicación y las dependencias:

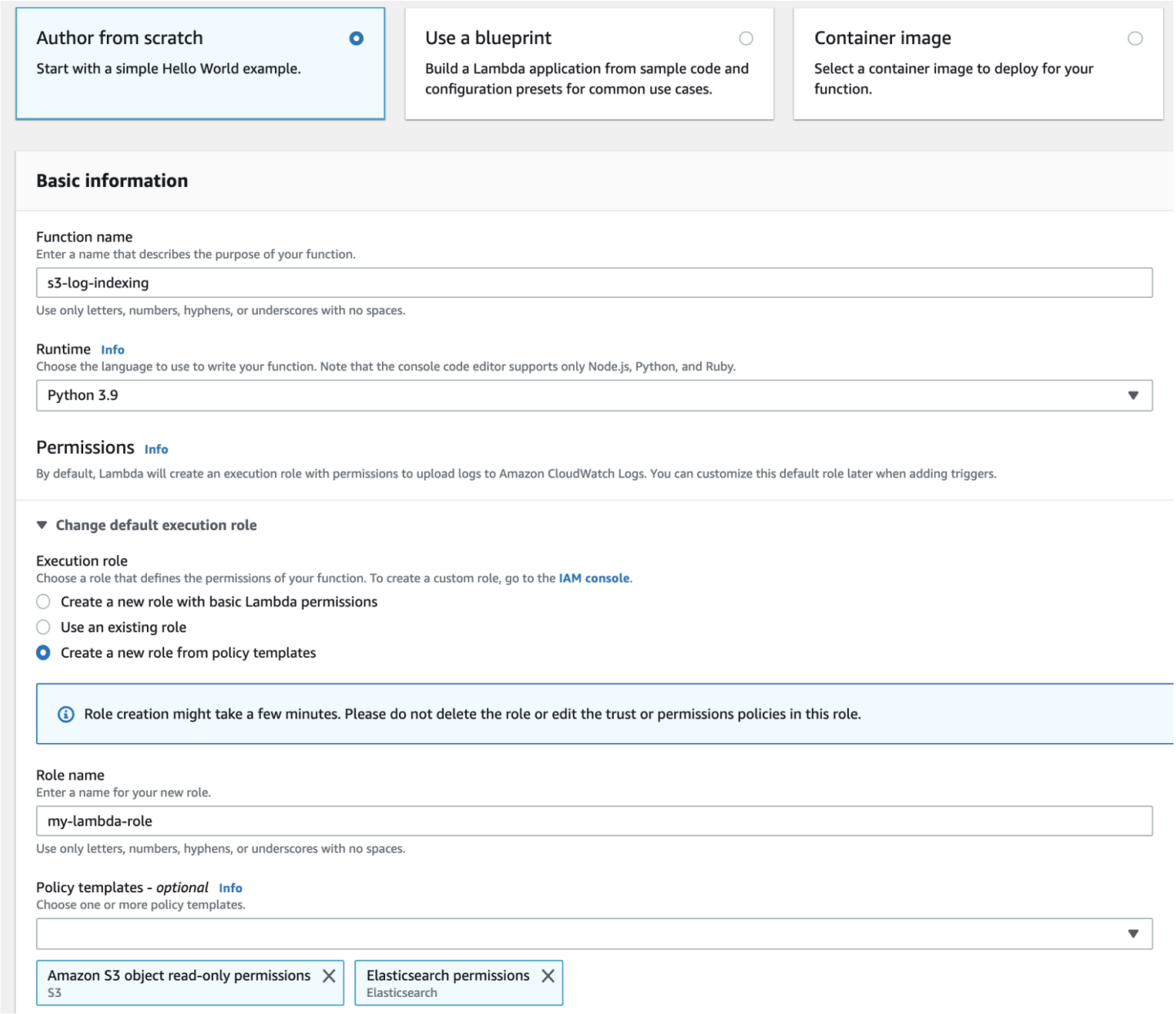
```
cd package
zip -r ../lambda.zip .
```

```
cd ..
zip -g lambda.zip sample.py
```

Crear la función de Lambda

Después de crear el paquete de implementación, puede crear la función de Lambda. Al crear una función, elija un nombre, un tiempo de ejecución (por ejemplo, Python 3.8) y un rol de IAM. El rol de IAM define los permisos para la función. Para obtener instrucciones detalladas, consulte Create a Lambda function with the consol en la Guía para desarrolladores de AWS Lambda.

En este ejemplo, se supone que utiliza la consola. Seleccione Python 3.9 y un rol que tenga permisos de lectura en S3 y permisos de escritura en OpenSearch Service, tal y como se muestra en la siguiente captura de pantalla:



Después de crear la función, debe agregar un desencadenador. En este ejemplo, queremos que el código se ejecute cada vez que llega un archivo de registros a un bucket de S3:

1. Seleccione Agregar desencadenador y Seleccione S3.
2. Seleccione el bucket.
3. Para el Tipo de evento, elija PUT.
4. Para el Prefijo, escriba `logs/`.
5. Para el Sufijo, escriba `.log`.
6. Reconozca la advertencia de invocación recursiva y elija Agregar.

Por último, puede cargar el paquete de implementación:

1. Seleccione Cargar desde y archivo .zip, luego siga las instrucciones para cargar el paquete de implementación.
2. Una vez finalizada la carga, edite la Configuración de tiempo de ejecución y cambie el Controlador a `sample.handler`. Esta configuración indica a Lambda el archivo (`sample.py`) y el método (`handler`) que debe ejecutar cuando se produzca un desencadenador.

En este momento, dispone de un conjunto completo de recursos: un bucket para los archivos de registro, una función que se ejecuta cada vez que se agrega un archivo de registros al bucket, el código que ejecuta el análisis y la indexación y un dominio de OpenSearch Service para la búsqueda y la visualización.

Prueba de la función de Lambda

Después de crear la función, puede probarla mediante la carga de un archivo en el bucket de Amazon S3. Cree un archivo denominado `sample.log` que contenga los siguientes ejemplos de líneas de registro:

```
12.345.678.90 - [10/Oct/2000:13:55:36 -0700] "PUT /some-file.jpg"
12.345.678.91 - [10/Oct/2000:14:56:14 -0700] "GET /some-file.jpg"
```

Cargue el archivo en la carpeta `logs` del bucket de S3. Para obtener instrucciones, consulte Carga de un objeto en el bucket en la Guía del usuario de Amazon Simple Storage Service.

A continuación, utilice la consola de OpenSearch Service o los OpenSearch Dashboards para verificar que el índice `lambda-s3-index` contiene dos documentos. También puede realizar una petición de búsqueda estándar:

```
GET https://domain-name/lambda-s3-index/_search?pretty
{
  "hits" : {
    "total" : 2,
    "max_score" : 1.0,
    "hits" : [
      {
        "_index" : "lambda-s3-index",
        "_type" : "_doc",
        "_id" : "vTYXaWIBJWV_TTkEuSDg",
```

```
        "_score" : 1.0,
        "_source" : {
          "ip" : "12.345.678.91",
          "message" : "GET /some-file.jpg",
          "timestamp" : "10/Oct/2000:14:56:14 -0700"
        }
      },
      {
        "_index" : "lambda-s3-index",
        "_type" : "_doc",
        "_id" : "vjYmaWIBJWV_TTkEuCAB",
        "_score" : 1.0,
        "_source" : {
          "ip" : "12.345.678.90",
          "message" : "PUT /some-file.jpg",
          "timestamp" : "10/Oct/2000:13:55:36 -0700"
        }
      }
    ]
  }
}
```

Cargar datos de streaming desde Amazon Kinesis Data Streams

Puede cargar datos de streaming desde Kinesis Data Streams en OpenSearch Service. Cuando llegan datos nuevos al flujo de datos, activan una notificación de eventos en Lambda que, a su vez, ejecuta el código personalizado para realizar la indexación. En esta sección, se incluye un código de muestra simple de Python.

Requisitos previos

Antes de continuar, debe contar con los siguientes recursos.

Requisito previo	Descripción
Amazon Kinesis Data Streams	Fuente de eventos de la función Lambda Para más información, consulte Kinesis Data Streams.
Dominio de OpenSearch Service	Es el destino de los datos después de que la función Lambda los procesa. Para más información, consulte the section called “ Creación de dominios de OpenSearch Service”.

Requisito previo	Descripción

Rol de IAM	Este rol debe tener permisos básicos para OpenSearch Service, Kinesis y Lambda, como los siguientes:
	
	JSON
	
	`{ "Version":"2012-10-17", "Statement": [ { "Effect": "Allow", "Action": [ "es:ESHttpPost", "es:ESHttpPut", "logs:CreateLogGroup", "logs:CreateLogStream", "logs:PutLogEvents", "kinesis:GetShardIterator", "kinesis:GetRecords", "kinesis:DescribeStream", "kinesis:ListStreams" ], "Resource": "*" } ] }`
	
	El rol debe tener la siguiente relación de confianza:
	
	JSON
	
	`{ "Version":"2012-10-17", "Statement": [ { "Effect": "Allow", "Principal": { "Service": "lambda.amazonaws.com" },`

Requisito previo	Descripción

	<pre>"Action": "sts:AssumeRole"
	}
	]
	}</pre>
	Para más información, consulte Creación de roles de IAM en la Guía del usuario de IAM.

Crear la función de Lambda

Siga las instrucciones de the section called “Crear el paquete de implementación de Lambda”, pero cree un directorio denominado `kinesis-to-opensearch` y utilice el siguiente código para `sample.py`:

```
import base64
import boto3
import json
import requests
from requests_aws4auth import AWS4Auth

region = '' # e.g. us-west-1
service = 'es'
credentials = boto3.Session().get_credentials()
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key, region, service,
 session_token=credentials.token)

host = '' # the OpenSearch Service domain, e.g. https://search-mydomain.us-
west-1.es.amazonaws.com
index = 'lambda-kine-index'
datatype = '_doc'
url = host + '/' + index + '/' + datatype + '/'

headers = { "Content-Type": "application/json" }

def handler(event, context):
    count = 0
    for record in event['Records']:
        id = record['eventID']
        timestamp = record['kinesis']['approximateArrivalTimestamp']
```

```
        # Kinesis data is base64-encoded, so decode here
        message = base64.b64decode(record['kinesis']['data'])

        # Create the JSON document
        document = { "id": id, "timestamp": timestamp, "message": message }
        # Index the document
        r = requests.put(url + id, auth=awsauth, json=document, headers=headers)
        count += 1
    return 'Processed ' + str(count) + ' items.'
```

Edite las variables para `region` y `host`.

Instale pip, si todavía no lo hizo, luego utilice los siguientes comandos para instalar las dependencias:

```
cd kinesis-to-opensearch

pip install --target ./package requests
pip install --target ./package requests_aws4auth
```

A continuación, siga las instrucciones de the section called “Crear la función de Lambda”, pero especifique el rol de IAM de the section called “Requisitos previos” y la configuración siguiente para el desencadenador:

- Flujo de Kinesis: su flujo de Kinesis
- Tamaño del lote: 100
- Posición inicial: Trim horizon

Para más información, consulte ¿Qué es Amazon Kinesis Data Streams? en la Guía para desarrolladores de Amazon Kinesis Data Streams.

En este momento dispone de un conjunto completo de recursos: un flujo de datos de Kinesis, una función que se ejecuta cuando se reciben datos nuevos en el flujo y se indexan y un dominio de OpenSearch Service para la búsqueda y la visualización.

Prueba de la función de Lambda

Después de crear la función, puede probarla al agregar un registro al flujo de datos mediante la AWS CLI:

```
aws kinesis put-record --stream-name test --data "My test data." --partition-key
 partitionKey1 --region us-west-1
```

A continuación, utilice la consola de OpenSearch Service o los OpenSearch Dashboards para verificar que `lambda-kine-index` contiene un documento. También puede utilizar la solicitud siguiente:

```
GET https://domain-name/lambda-kine-index/_search
{
  "hits" : [
    {
      "_index": "lambda-kine-index",
      "_type": "_doc",
      "_id":
 "shardId-000000000000:495835116157626994950129608214214566865294366804960870 42",
      "_score": 1,
      "_source": {
        "timestamp": 1523648740.051,
        "message": "My test data.",
        "id":
 "shardId-000000000000:495835116157626994950129608214214566865294366804960870 42"
      }
    }
  ]
}
```

Carga de datos de streaming desde Amazon DynamoDB

Puede utilizar AWS Lambda para enviar datos al dominio de OpenSearch Service desde Amazon DynamoDB. Cuando llegan datos nuevos a la tabla de base de datos, activan una notificación de eventos en Lambda que, a su vez, ejecuta el código personalizado para realizar la indexación.

Requisitos previos

Antes de continuar, debe contar con los siguientes recursos.

Requisito previo	Descripción
Tabla de DynamoDB	La tabla contiene los datos de origen. Para más información, consulte Operaciones básicas en tablas de DynamoDB en la Guía para desarroll adores de Amazon DynamoDB.

Requisito previo	Descripción
	La tabla debe encontrarse en la misma región que el dominio de OpenSearch Service y debe tener un flujo configurado como Nueva imagen. Para más información, consulte Habilitación de un flujo.
Dominio de OpenSearch Service	Es el destino de los datos después de que la función de Lambda los procesa. Para más información, consulte the section called “ Creación de dominios de OpenSearch Service”.

Requisito previo	Descripción
rol de IAM	Este rol debe tener permisos de ejecución básicos para OpenSearch Service, DynamoDB y Lambda, como los siguientes: JSON `{ "Version":"2012-10-17", "Statement": [{ "Effect": "Allow", "Action": ["es:ESHttpPost", "es:ESHttpPut", "dynamodb:DescribeStream", "dynamodb:GetRecords", "dynamodb:GetShardIterator", "dynamodb:ListStreams", "logs:CreateLogGroup", "logs:CreateLogStream", "logs:PutLogEvents"], "Resource": "*" }] }`  El rol debe tener la siguiente relación de confianza:  JSON  `{ "Version":"2012-10-17", "Statement": [{ "Effect": "Allow", "Principal": { "Service": "lambda.amazonaws.com" },`

Requisito previo	Descripción

	<pre>"Action": "sts:AssumeRole"
	}
	]
	}</pre>
	Para más información, consulte Creación de roles de IAM en la Guía del usuario de IAM.

Crear la función de Lambda

Siga las instrucciones de the section called “Crear el paquete de implementación de Lambda”, pero cree un directorio denominado `ddb-to-opensearch` y utilice el siguiente código para `sample.py`:

```
import boto3
import requests
from requests_aws4auth import AWS4Auth

region = '' # e.g. us-east-1
service = 'es'
credentials = boto3.Session().get_credentials()
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key, region, service,
 session_token=credentials.token)

host = '' # the OpenSearch Service domain, e.g. https://search-mydomain.us-
west-1.es.amazonaws.com
index = 'lambda-index'
datatype = '_doc'
url = host + '/' + index + '/' + datatype + '/'

headers = { "Content-Type": "application/json" }

def handler(event, context):
    count = 0
    for record in event['Records']:
        # Get the primary key for use as the OpenSearch ID
        id = record['dynamodb']['Keys']['id']['S']

        if record['eventName'] == 'REMOVE':
            r = requests.delete(url + id, auth=awsauth)
```

```
        else:
            document = record['dynamodb']['NewImage']
            r = requests.put(url + id, auth=awsauth, json=document, headers=headers)
        count += 1
    return str(count) + ' records processed.'
```

Edite las variables para `region` y `host`.

Instale pip, si todavía no lo hizo, luego utilice los siguientes comandos para instalar las dependencias:

```
cd ddb-to-opensearch

pip install --target ./package requests
pip install --target ./package requests_aws4auth
```

A continuación, siga las instrucciones de the section called “Crear la función de Lambda”, pero especifique el rol de IAM de the section called “Requisitos previos” y la configuración siguiente para el desencadenador:

- Tabla: tabla de DynamoDB
- Tamaño del lote: 100
- Posición inicial: Trim horizon

Para más información, consulte Process New Items with DynamoDB Streams and Lambda en la Guía para desarrolladores de Amazon DynamoDB.

En este momento, dispone de un conjunto completo de recursos: una tabla de DynamoDB para los datos fuente, un flujo de DynamoDB de cambios en la tabla, una función que se ejecuta cuando cambian los datos fuente y se indexan y un dominio de OpenSearch Service para la búsqueda y la visualización.

Prueba de la función de Lambda

Después de crear la función, puede probarla al agregar un elemento a la tabla de DynamoDB mediante la AWS CLI:

```
aws dynamodb put-item --table-name test --item '{"director": {"S": "Kevin
 Costner"},"id": {"S": "00001"},"title": {"S": "The Postman"}}' --region us-west-1
```

A continuación, utilice la consola de OpenSearch Service o los OpenSearch Dashboards para verificar que `lambda-index` contiene un documento. También puede utilizar la solicitud siguiente:

```
GET https://domain-name/lambda-index/_doc/00001
{
    "_index": "lambda-index",
    "_type": "_doc",
    "_id": "00001",
    "_version": 1,
    "found": true,
    "_source": {
        "director": {
            "S": "Kevin Costner"
        },
        "id": {
            "S": "00001"
        },
        "title": {
            "S": "The Postman"
        }
    }
}
```

Carga de datos de streaming desde Amazon Data Firehose

Firehose es compatible con OpenSearch Service como destino de entrega. Para obtener instrucciones acerca de cómo cargar los datos de streaming en OpenSearch Service, consulte Creating a Kinesis Data Firehose Delivery Stream y Choose OpenSearch Service for Your Destination en la Guía para desarrolladores de Amazon Data Firehose.

Es posible que necesite realizar transformaciones en los datos antes de cargarlos en OpenSearch Service. Para obtener más información acerca de cómo utilizar las funciones Lambda para realizar esta tarea, consulte Transformación de datos en Amazon Kinesis Data Firehose en la misma guía.

Al configurar un flujo de entrega, Firehose ofrece un rol de IAM de “un clic” que ofrece acceso a los recursos que necesita para enviar datos a OpenSearch Service, hacer copias de seguridad de los datos en Amazon S3 y transformar los datos con Lambda. Debido a la complejidad que supone la creación de ese rol manualmente, recomendamos utilizar el rol proporcionado.

Cargar datos de streaming desde Amazon CloudWatch

Puede cargar datos de streaming desde CloudWatch Logs en el dominio de OpenSearch Service mediante una suscripción a Registros de CloudWatch. Para obtener más información sobre las suscripciones a Amazon CloudWatch, consulte Procesamiento en tiempo real de datos de registros con suscripciones. Para obtener información sobre la configuración, consulte Transmitir datos de CloudWatch Logs a Amazon OpenSearch Service en la Guía para desarrolladores de Amazon CloudWatch.

Carga de datos de streaming desde AWS IoT

Puede enviar datos desde AWS IoT mediante reglas. Para más información, consulte la acción de OpenSearch en la Guía para desarrolladores de AWS IoT.

Carga de datos en Amazon OpenSearch Service con Logstash

La versión de código abierto de Logstash (Logstash OSS) proporciona una forma conveniente de utilizar la API masiva para cargar datos en un dominio de Amazon OpenSearch Service. El servicio admite todos los complementos de entrada de Logstash estándar, incluido el complemento de entrada de Amazon S3. OpenSearch Service soporta el complemento de salida logstash-output-opensearch, que soporta autenticación básica y credenciales de IAM. El complemento funciona con la versión 8.1 y versiones inferiores de Logstash OSS.

Configuración

La configuración de Logstash varía en función del tipo de autenticación que utiliza su dominio.

Independientemente del método de autenticación que utilice, debe configurar `ecs_compatibility` en `disabled` en la sección de salida del archivo de configuración. Logstash 8.0 introdujo un cambio revolucionario en el que se ejecutan todos los complementos en Modo de compatibilidad ECS de forma predeterminada. Debe anular el valor predeterminado para mantener el comportamiento heredado.

Configuración de control de acceso detallado

Si el dominio de OpenSearch Service utiliza un control de acceso detallado con autenticación HTTP básica, la configuración es similar a cualquier otro clúster de OpenSearch. Este archivo de configuración de ejemplo toma su entrada de la versión de código abierto de Filebeat (Filebeat OSS):

```
input {
  beats  {
    port => 5044
  }
}

output {
  opensearch {
    hosts       => "https://domain-endpoint:443"
    user        => "my-username"
    password    => "my-password"
    index       => "logstash-logs-%{+YYYY.MM.dd}"
    ecs_compatibility => disabled
    ssl_certificate_verification => false
  }
}
```

La configuración varía según la aplicación de Beats y el caso de uso, pero la configuración de Filebeat OSS podría verse así:

```
filebeat.inputs:
- type: log
  enabled: true
  paths:
    - /path/to/logs/dir/*.log
filebeat.config.modules:
  path: ${path.config}/modules.d/*.yml
  reload.enabled: false
setup.ilm.enabled: false
setup.ilm.check_exists: false
setup.template.settings:
  index.number_of_shards: 1
output.logstash:
  hosts: ["logstash-host:5044"]
```

Configuración de IAM

Si el dominio utiliza una política de acceso de dominio basada en IAM o un control de acceso detallado con un usuario maestro, debe firmar todas las solicitudes de OpenSearch Service con las credenciales de IAM. La siguiente política basada en identidades concede todas las solicitudes HTTP a los subrecursos de su dominio.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "es:ESHttp*"
      ],
      "Resource": "arn:aws:es:us-east-1:111122223333:domain/domain-name/*"
    }
  ]
}
```

Para configurar su configuración de Logstash, cambie su archivo de configuración para usar el complemento en su salida. Este ejemplo de archivo de configuración toma su entrada de archivos en un bucket de S3:

```
input {
  s3 {
    bucket => "amzn-s3-demo-"
    region => "us-east-1"
  }
}

output {
  opensearch {
    hosts => ["domain-endpoint:443"]
    auth_type => {
      type => 'aws_iam'
      aws_access_key_id => 'your-access-key'
      aws_secret_access_key => 'your-secret-key'
      region => 'us-east-1'
      }
      index  => "logstash-logs-%{+YYYY.MM.dd}"
      ecs_compatibility => disabled
  }
}
```

Si no desea proporcionar sus credenciales de IAM dentro del archivo de configuración, puede exportarlas (o ejecutarlas `aws configure`):

```
export AWS_ACCESS_KEY_ID="your-access-key"
export AWS_SECRET_ACCESS_KEY="your-secret-key"
export AWS_SESSION_TOKEN="your-session-token"
```

Si el dominio de OpenSearch Service está en una VPC, la máquina de Logstash OSS debe poder conectarse a la VPC y tener acceso al dominio a través de los grupos de seguridad de la VPC. Para más información, consulte the section called “Acerca de las políticas de acceso en los dominios de VPC”.

Buscar datos en Amazon OpenSearch Service

Existen varios métodos habituales de búsqueda de documentos en Amazon OpenSearch Service, incluidas las búsquedas de URI y las búsquedas de cuerpos de la solicitud. OpenSearch Service ofrece funcionalidad adicional que mejora la experiencia de búsqueda, como paquetes personalizados, compatibilidad con SQL y búsqueda asincrónica. Para ver una referencia de la API de búsqueda de OpenSearch completa, consulte la documentación de OpenSearch.

> **Note**
>
> Las siguientes solicitudes de muestra funcionan con las API de OpenSearch. Algunas solicitudes podrían no funcionar con versiones anteriores de Elasticsearch.

Temas

- Búsquedas de URI
- Búsquedas de cuerpo de la solicitud
- Paginar los resultados de búsqueda
- Lenguaje de consulta de paneles
- Importación y administración de paquetes en Amazon OpenSearch Service
- Consultar los datos de Amazon OpenSearch Service con SQL
- Búsqueda en clústeres de Amazon OpenSearch Service
- Aprender a clasificar para Amazon OpenSearch Service
- Búsqueda asíncrona en Amazon OpenSearch Service
- Búsqueda de punto en el tiempo en Amazon OpenSearch Service
- Búsqueda semántica en Amazon OpenSearch Service
- Búsqueda de segmentos simultánea en Amazon OpenSearch Service
- Generación de consultas en lenguaje natural en Amazon OpenSearch Service

Búsquedas de URI

Las búsquedas de identificador de recursos universal (URI) son la forma más sencilla de búsqueda. En una búsqueda de URI, hay que especificar la consulta como un parámetro de solicitud HTTP:

```
GET https://search-my-domain.us-west-1.es.amazonaws.com/_search?q=house
```

Una respuesta de muestra tendría el siguiente aspecto:


```
{
  "took": 25,
  "timed_out": false,
  "_shards": {
    "total": 10,
    "successful": 10,
    "skipped": 0,
    "failed": 0
  },
  "hits": {
    "total": {
      "value": 85,
      "relation": "eq",
    },
    "max_score": 6.6137657,
    "hits": [
      {
        "_index": "movies",
        "_type": "movie",
        "_id": "tt0077975",
        "_score": 6.6137657,
        "_source": {
          "directors": [
            "John Landis"
          ],
          "release_date": "1978-07-27T00:00:00Z",
          "rating": 7.5,
          "genres": [
            "Comedy",
            "Romance"
          ],
          "image_url": "http://ia.media-imdb.com/images/M/
MV5BMTY2OTQxNTc1OF5BMl5BanBnXkFtZTYwNjA3NjI5._V1_SX400_.jpg",
          "plot": "At a 1962 College, Dean Vernon Wormer is determined to expel the
 entire Delta Tau Chi Fraternity, but those troublemakers have other plans for him.",
          "title": "Animal House",
          "rank": 527,
          "running_time_secs": 6540,
          "actors": [
```

```
            "John Belushi",
            "Karen Allen",
            "Tom Hulce"
          ],
          "year": 1978,
          "id": "tt0077975"
        }
      },
      ...
    ]
  }
}
```

De forma predeterminada, esta consulta realiza búsquedas en todos los campos de todos los índices para encontrar el término house. Para aportar más precisión a la búsqueda, especifique un índice (`movies`) y un campo de documento (`title`) en el URI:

```
GET https://search-my-domain.us-west-1.es.amazonaws.com/movies/_search?q=title:house
```

Puede incluir parámetros adicionales en la solicitud, pero los parámetros compatibles proporcionan solo un pequeño subconjunto de opciones de búsqueda de OpenSearch. La siguiente solicitud devuelve 20 resultados (en vez de los 10 predeterminados) y los clasifica por año (en vez de clasificarlos por `_score`):

```
GET https://search-my-domain.us-west-1.es.amazonaws.com/movies/_search?
q=title:house&size=20&sort=year:desc
```

Búsquedas de cuerpo de la solicitud

A fin de realizar búsquedas más complejas, utilice el cuerpo de la solicitud HTTP y el lenguaje específico de dominio (DSL) de OpenSearch para realizar las consultas. El DSL de la consulta permite especificar la gama completa de opciones de búsqueda de OpenSearch.

> **Note**
>
> No puede incluir caracteres especiales Unicode en el valor de un campo de texto, o el valor se analizará como varios valores separados por el carácter especial. Este análisis incorrecto puede provocar un filtrado involuntario de los documentos y comprometer potencialmente

el control sobre su acceso. Para más información, consulte A note on Unicode special characters in text fields de la documentación de OpenSearch.

La siguiente consulta de `match` es similar al ejemplo de búsqueda de URI final:

```
POST https://search-my-domain.us-west-1.es.amazonaws.com/movies/_search
{
  "size": 20,
  "sort": {
    "year": {
      "order": "desc"
    }
  },
  "query": {
    "query_string": {
      "default_field": "title",
      "query": "house"
    }
  }
}
```

Note

La API de `_search` acepta `GET` y `POST` HTTP para búsquedas del cuerpo de la solicitud, pero no todos los clientes HTTP admiten la adición de un cuerpo de la solicitud a una solicitud `GET`. `POST` es la opción más universal.

En muchos casos, es posible que desee buscar varios campos, pero no todos los campos. Utilice la consulta `multi_match`:

```
POST https://search-my-domain.us-west-1.es.amazonaws.com/movies/_search
{
  "size": 20,
  "query": {
    "multi_match": {
      "query": "house",
      "fields": ["title", "plot", "actors", "directors"]
    }
```

```
  }
}
```

Potenciar campos

Puede mejorar la relevancia de la búsqueda mediante la "potenciación" de ciertos campos. Los potenciadores son multiplicadores que dan mayor peso a las coincidencias en un campo que a las de otros campos. En el siguiente ejemplo, una coincidencia de john en el campo `title` tiene el doble de peso en `_score` que una coincidencia en el campo `plot` y cuatro veces más que otra en los campos `actors` o `directors`. El resultado es que películas como John Wick y John Carter están cerca de los primeros resultados de búsqueda y las películas protagonizadas por John Travolta están casi al final.

```
POST https://search-my-domain.us-west-1.es.amazonaws.com/movies/_search
{
  "size": 20,
  "query": {
    "multi_match": {
      "query": "john",
      "fields": ["title^4", "plot^2", "actors", "directors"]
    }
  }
}
```

Resaltado de resultados de búsqueda

La opción `highlight` indica a OpenSearch que devuelva un objeto adicional dentro de la matriz `hits` si la consulta coincide con uno o más campos:

```
POST https://search-my-domain.us-west-1.es.amazonaws.com/movies/_search
{
  "size": 20,
  "query": {
    "multi_match": {
      "query": "house",
      "fields": ["title^4", "plot^2", "actors", "directors"]
    }
  },
  "highlight": {
    "fields": {
```

```
      "plot": {}
    }
  }
}
```

Si la consulta coincide con el contenido del campo `plot`, un acierto podría tener el siguiente aspecto:

```
{
  "_index": "movies",
  "_type": "movie",
  "_id": "tt0091541",
  "_score": 11.276199,
  "_source": {
    "directors": [
      "Richard Benjamin"
    ],
    "release_date": "1986-03-26T00:00:00Z",
    "rating": 6,
    "genres": [
      "Comedy",
      "Music"
    ],
    "image_url": "http://ia.media-imdb.com/images/M/
MV5BMTIzODEzODE2OF5BMl5BanBnXkFtZTcwNjQ3ODcyMQ@@._V1_SX400_.jpg",
    "plot": "A young couple struggles to repair a hopelessly dilapidated house.",
    "title": "The Money Pit",
    "rank": 4095,
    "running_time_secs": 5460,
    "actors": [
      "Tom Hanks",
      "Shelley Long",
      "Alexander Godunov"
    ],
    "year": 1986,
    "id": "tt0091541"
  },
  "highlight": {
    "plot": [
      "A young couple struggles to repair a hopelessly dilapidated <em>house</em>."
    ]
  }
}
```

De forma predeterminada, OpenSearch incluye la cadena coincidente entre etiquetas , proporciona hasta 100 caracteres de contexto en torno a la coincidencia y desglosa el contenido en frases mediante la identificación de signos de puntuación, espacios, tabulaciones y saltos de línea. Todas estas configuraciones son personalizables:

```
POST https://search-my-domain.us-west-1.es.amazonaws.com/movies/_search
{
  "size": 20,
  "query": {
    "multi_match": {
      "query": "house",
      "fields": ["title^4", "plot^2", "actors", "directors"]
    }
  },
  "highlight": {
    "fields": {
      "plot": {}
    },
    "pre_tags": "<strong>",
    "post_tags": "</strong>",
    "fragment_size": 200,
    "boundary_chars": ".,!? "
  }
}
```

API de recuento

Si no está interesado en el contenido de sus documentos y solo quiere saber el número de coincidencias, puede utilizar la API de `_count` en lugar de la API de `_search`. En la siguiente solicitud, se utiliza la consulta `query_string` para identificar comedias románticas:

```
POST https://search-my-domain.us-west-1.es.amazonaws.com/movies/_count
{
  "query": {
    "query_string": {
      "default_field": "genres",
      "query": "romance AND comedy"
    }
  }
}
```

Una respuesta de muestra tendría el siguiente aspecto:

```
{
  "count": 564,
  "_shards": {
    "total": 5,
    "successful": 5,
    "skipped": 0,
    "failed": 0
  }
}
```

Paginar los resultados de búsqueda

Si necesita mostrar un gran número de resultados de búsqueda, puede implementar la paginación usando varios métodos diferentes.

Punto en el tiempo

La característica de punto en el tiempo (PIT) es un tipo de búsqueda que permite ejecutar diferentes consultas en un conjunto de datos fijo en el tiempo. Este es el método de paginación preferido en OpenSearch, especialmente para la paginación profunda. Puede usar PIT con la versión 2.5 y versiones posteriores de OpenSearch Service. Para más información sobre PIT, consulte ???.

Los parámetros **`from`** y **`size`**

La forma más sencilla de paginar es con los parámetros `from` y `size`. La siguiente solicitud devuelve los resultados 20-39 de la lista de resultados de búsqueda indexada a cero:

```
POST https://search-my-domain.us-west-1.es.amazonaws.com/movies/_search
{
  "from": 20,
  "size": 20,
  "query": {
    "multi_match": {
      "query": "house",
      "fields": ["title^4", "plot^2", "actors", "directors"]
    }
  }
}
```

Para ver más información sobre la paginación de búsquedas, consulte Paginar resultados en la documentación de OpenSearch.

Lenguaje de consulta de paneles

Puede utilizar el Lenguaje de consulta de paneles (DQL) para buscar datos y visualizaciones en los paneles de OpenSearch. DQL utiliza cuatro tipos de consultas principales: términos, booleano, fecha e intervalo y campo anidado.

Consulta de términos

Una consulta de términos requiere que se especifique el término que se busca.

Para llevar a cabo una consulta de términos, ingrese lo siguiente:

```
host:www.example.com
```

Consulta booleana

Puede utilizar los operadores booleanos `AND`, `or` y `not` para combinar varias consultas.

Para llevar a cabo una consulta booleana, pegue lo siguiente:

```
host.keyword:www.example.com and response.keyword:200
```

Consulta de fecha e intervalo

Puede utilizar una consulta de fecha e intervalo para buscar una fecha anterior o posterior a la consulta.

- > indica una búsqueda de una fecha posterior a la fecha especificada.
- < indica una búsqueda de una fecha anterior a la fecha especificada.

```
@timestamp > "2020-12-14T09:35:33"
```

Consulta de campo anidada

Si tiene un documento con campos anidados, debe especificar qué partes del documento desea recuperar. A continuación, se muestra un documento de muestra con campos anidados:

```
{"NBA players":[
    {"player-name": "Lebron James",
      "player-position": "Power forward",
      "points-per-game": "30.3"
    },
    {"player-name": "Kevin Durant",
      "player-position": "Power forward",
      "points-per-game": "27.1"
    },
    {"player-name": "Anthony Davis",
      "player-position": "Power forward",
      "points-per-game": "23.2"
    },
    {"player-name": "Giannis Antetokounmpo",
      "player-position": "Power forward",
      "points-per-game":"29.9"
    }
  ]
}
```

Para recuperar un campo específico con DQL, pegue lo siguiente:

```
NBA players: {player-name: Lebron James}
```

Para recuperar varios objetos del documento anidado, pegue lo siguiente:

```
NBA players: {player-name: Lebron James} and NBA players: {player-name: Giannis
 Antetokounmpo}
```

Para buscar dentro de un intervalo, pegue lo siguiente:

```
NBA players: {player-name: Lebron James} and NBA players: {player-name: Giannis
 Antetokounmpo and < 30}
```

Si el documento tiene un objeto anidado dentro de otro objeto, aún puede recuperar datos especificando todos los niveles. Para ello, pegue lo siguiente:

```
Top-Power-forwards.NBA players: {player-name:Lebron James}
```

Importación y administración de paquetes en Amazon OpenSearch Service

Amazon OpenSearch Service le permite cargar archivos de diccionario personalizados, como listas de palabras vacías (stop words) y sinónimos y asociar complementos con su dominio. Estos complementos pueden ser preempaquetados, personalizados o de terceros, lo que le brinda flexibilidad para ampliar la funcionalidad de su dominio. El término genérico para referirse a todos estos tipos de archivos es paquetes.

- Los archivos de diccionario ayudan a perfeccionar los resultados de búsqueda al indicar a OpenSearch que ignore palabras comunes de alta frecuencia o que trate términos similares, como “crema helada”, “helado” y “sorbete” como equivalentes. También pueden mejorar la derivación, como se observa con el complemento de análisis japonés (kuromoji).
- Los complementos preempaquetados ofrecen funciones integradas, como el complemento Amazon Personalize para obtener resultados de búsqueda personalizados. Estos complementos utilizan el tipo de paquete `ZIP-PLUGIN`. Para obtener más información, consulte the section called “Complementos por versión de motor”.
- Los complementos personalizados y de terceros le permiten agregar características personalizadas o integrarlos con sistemas externos, lo que ofrece aún más flexibilidad para su dominio. Al igual que los complementos preempaquetados, sube los complementos personalizados como paquetes `ZIP-PLUGIN`. En el caso de los complementos de terceros, también debe importar los archivos de licencia y configuración del complemento como paquetes independientes y, a continuación, asociarlos todos al dominio.

  Para obtener más información, consulte los temas siguientes:

  - the section called “Complementos personalizados”
  - the section called “Complementos de terceros”

> **Note**
>
> Puede asociar un máximo de 20 complementos con un único dominio. Este límite incluye todos los tipos de complementos: opcionales, de terceros y personalizados.

Temas

- Permisos necesarios
- Carga de paquetes en Amazon S3
- Importación y asociación de paquetes
- Uso de paquetes con OpenSearch
- Actualización de paquetes
- Actualización manual de los índices con un nuevo diccionario
- Disociación y eliminación de paquetes
- Administración de complementos personalizados en Amazon OpenSearch Service
- Instalación de complementos de terceros en Amazon OpenSearch Service

Permisos necesarios

Los usuarios sin acceso de administrador requieren ciertas acciones de AWS Identity and Access Management (IAM) para administrar paquetes:

- `es:CreatePackage`: crear un paquete
- `es:DeletePackage`: eliminar un paquete
- `es:AssociatePackage`: asociar un paquete a un dominio
- `es:DissociatePackage`: disociar un paquete de un dominio

También necesita permisos en la ruta del bucket de Amazon S3 u objeto donde reside el paquete personalizado.

Otorgue todos los permisos en IAM, no en la política de acceso al dominio. Para más información, consulte the section called “Identity and Access Management”.

Carga de paquetes en Amazon S3

Esta sección explica cómo cargar paquetes de diccionarios personalizados, dado que los paquetes de complementos prediseñados ya están instalados. Antes de poder asociar un diccionario personalizado a un dominio, debe cargarlo en un bucket de Amazon S3. Para ver las instrucciones, consulte Carga de objetos en la Guía del usuario de Amazon Simple Storage Service. No es necesario cargar los complementos admitidos.

Si el diccionario contiene información confidencial, especifique el cifrado del servidor mediante claves administradas por S3 al cargarlo. El Servicio OpenSearch no puede acceder a archivos en S3 protegidos con una clave AWS KMS.

Después de cargar el archivo, tome nota de su ruta de S3. El formato de la ruta es `s3://amzn-s3-demo-bucket/file-path/file-name`.

Puede utilizar el siguiente archivo de sinónimos para fines de prueba. Guárdelo como `synonyms.txt`.

```
danish, croissant, pastry
ice cream, gelato, frozen custard
sneaker, tennis shoe, running shoe
basketball shoe, hightop
```

Algunos diccionarios, como los diccionarios Hunspell, utilizan varios archivos y requieren sus propios directorios en el sistema de archivos. En este momento, OpenSearch Service solo admite diccionarios de un solo archivo.

Importación y asociación de paquetes

La consola es la forma más sencilla de importar un diccionario personalizado a OpenSearch Service. Al importar un diccionario desde Amazon S3, OpenSearch Service almacena su propia copia del paquete y cifra automáticamente esa copia mediante AES-256 con claves administradas por OpenSearch Service.

Los complementos opcionales ya vienen preinstalados en OpenSearch Service, por lo que no es necesario que los cargue usted mismo, pero sí debe asociar un complemento a un dominio. Los complementos disponibles aparecen en la pantalla Paquetes de la consola.

Para importar y asociar un paquete a un dominio

1. En la consola de Amazon OpenSearch Service, seleccione Paquetes.
2. Elija Import package (Importar paquete).
3. Asigne al paquete un nombre descriptivo.
4. Proporcione la ruta de S3 al archivo y, a continuación, elija Importar.
5. Vuelva a la pantalla Paquetes.
6. Cuando el estado del paquete sea Disponible, elíjalo.

7. Seleccione Asociar a un dominio.
8. Seleccione un dominio y, a continuación, elija Siguiente. Revise los paquetes y seleccione Asociar.
9. En el panel de navegación, seleccione el dominio y, a continuación, vaya a la pestaña Paquetes.
10. Si el paquete es un diccionario personalizado, anote el ID cuando el paquete esté disponible. Utilice `analyzers/id` como ruta de acceso del archivo en las solicitudes a OpenSearch.

Uso de paquetes con OpenSearch

Esta sección explica cómo usar ambos tipos de paquetes: diccionarios personalizados y complementos prediseñados.

Uso de diccionarios personalizados

Después de asociar un archivo a un dominio, puede usarlo en parámetros como `synonyms_path`, `stopwords_path` y `user_dictionary` al crear generadores de tokens y filtros de token. El parámetro exacto varía según el objeto. Varios objetos admiten `synonyms_path` y `stopwords_path`, pero `user_dictionary` es exclusivo del complemento kuromoji.

Para el complemento IK (chino) Analysis, puede cargar un archivo de diccionario personalizado en forma de paquete personalizado y asociarlo a un dominio, y el complemento lo recogerá automáticamente sin necesidad de parámetro `user_dictionary`. Si se trata de un archivo de sinónimos, utilice el parámetro `synonyms_path`.

El siguiente ejemplo agrega un archivo de sinónimos a un nuevo índice:

```
PUT my-index
{
  "settings": {
    "index": {
      "analysis": {
        "analyzer": {
          "my_analyzer": {
            "type": "custom",
            "tokenizer": "standard",
            "filter": ["my_filter"]
          }
        },
```

```
        "filter": {
          "my_filter": {
            "type": "synonym",
            "synonyms_path": "analyzers/F111111111",
            "updateable": true
          }
        }
      }
    }
  },
  "mappings": {
    "properties": {
      "description": {
        "type": "text",
        "analyzer": "standard",
        "search_analyzer": "my_analyzer"
      }
    }
  }
}
```

Esta solicitud crea un analizador personalizado para el índice que utiliza el generador de tokens estándar y un filtro de token de sinónimos.

- Los generadores de tokens dividen las secuencias de caracteres en tokens (normalmente palabras) en función de algún conjunto de reglas. El ejemplo más simple es el generador de tokens de espacios en blanco, que divide los caracteres anteriores en un token cada vez que encuentra un carácter de espacio en blanco. Un ejemplo más complejo es el generador de tokens estándar, que utiliza un conjunto de reglas gramaticales para funcionar en muchos idiomas.
- Los filtros de tokens agregan, modifican o eliminan tokens. Por ejemplo, un filtro de token de sinónimos agrega tokens cuando encuentra una palabra en la lista de sinónimos. El filtro de tokens de palabras vacías elimina los tokens cuando encuentra una palabra en la lista de palabras vacías.

Esta solicitud también agrega un campo de texto (`description`) al mapeo e indica a OpenSearch que utilice el nuevo analizador como analizador de búsqueda. Puede ver que todavía utiliza el analizador estándar como analizador de índices.

Por último, tenga en cuenta la línea `"updateable": true` en el filtro de token. Este campo solo se aplica a los analizadores de búsqueda, no a los analizadores de índice, y es crítico si luego desea actualizar el analizador de búsqueda de forma automática.

Con fines de prueba, agregue algunos documentos al índice:

```
POST _bulk
{ "index": { "_index": "my-index", "_id": "1" } }
{ "description": "ice cream" }
{ "index": { "_index": "my-index", "_id": "2" } }
{ "description": "croissant" }
{ "index": { "_index": "my-index", "_id": "3" } }
{ "description": "tennis shoe" }
{ "index": { "_index": "my-index", "_id": "4" } }
{ "description": "hightop" }
```

A continuación, use un sinónimo para buscarlos:

```
GET my-index/_search
{
  "query": {
    "match": {
      "description": "gelato"
    }
  }
}
```

En este caso, OpenSearch devuelve la siguiente respuesta:

```
{
  "hits": {
    "total": {
      "value": 1,
      "relation": "eq"
    },
    "max_score": 0.99463606,
    "hits": [{
      "_index": "my-index",
      "_type": "_doc",
      "_id": "1",
      "_score": 0.99463606,
      "_source": {
        "description": "ice cream"
      }
    }]
  }
```

```
}
```

> **Tip**
>
> Los archivos de diccionario utilizan espacio de montón de Java proporcional a su tamaño. Por ejemplo, un archivo de diccionario de 2 GiB podría consumir 2 GiB de espacio de montón en un nodo. Si utiliza archivos de gran tamaño, asegúrese de que los nodos tengan suficiente espacio en el montón para acomodarlos. Monitoree la métrica `JVMMemoryPressure` y escale el clúster según sea necesario.

Uso de complementos preempaquetados

El servicio OpenSearch le permite asociar complementos de OpenSearch opcionales y preinstalados para usarlos con su dominio. Un paquete de complementos preempaquetados es compatible con una versión específica de OpenSearch y solo se puede asociar a los dominios con esa versión. La lista de paquetes disponibles para su dominio incluye todos los complementos compatibles con la versión de su dominio. Después de asociar un complemento a un dominio, comienza un proceso de instalación en el dominio. Luego, puede hacer referencia al complemento y usarlo cuando realice solicitudes a OpenSearch Service.

Asociar y disociar un complemento requiere una implementación azul/verde. Para más información, consulte the section called “Cambios que suelen causar implementaciones azul/verde”.

Los complementos opcionales incluyen analizadores de idioma y resultados de búsqueda personalizados. Por ejemplo, el complemento Amazon Personalize Search Ranking utiliza machine learning para personalizar los resultados de búsqueda para sus clientes. Para obtener más información sobre este complemento, consulte Personalización de los resultados de búsqueda desde OpenSearch. Para ver una lista de todos los complementos admitidos, consulte the section called “Complementos por versión de motor”.

Complemento Sudachi

En el caso del complemento Sudachi, al volver a asociar un archivo de diccionario, no se refleja inmediatamente en el dominio. El diccionario se actualiza cuando se ejecuta la siguiente implementación azul/verde en el dominio como parte de un cambio de configuración u otra actualización. Como alternativa, puede crear un nuevo paquete con los datos actualizados, crear un nuevo índice con un nuevo paquete, volver a indexar el índice existente en el nuevo índice y, a

continuación, eliminar el índice anterior. Si prefiere utilizar el enfoque de reindexación, utilice un alias de índice para que no se interrumpa el tráfico.

Además, el complemento Sudachi solo admite diccionarios binarios de Sudachi, que puede cargar con la operación de API CreatePackage. Para obtener información sobre el diccionario del sistema prediseñado y el proceso de compilación de diccionarios de usuario, consulte Documentación de Sudachi.

En el siguiente ejemplo, se muestra cómo utilizar los diccionarios de sistema y de usuario con el tokenizador de Sudachi. Debe cargar estos diccionarios como paquetes personalizados con el tipo `TXT-DICTIONARY` y proporcionar sus ID de paquete en la configuración adicional.

```
PUT sudachi_sample
{
  "settings": {
    "index": {
      "analysis": {
        "tokenizer": {
          "sudachi_tokenizer": {
            "type": "sudachi_tokenizer",
            "additional_settings": "{\"systemDict\": \"<system-dictionary-package-id>\",\"userDict\": [\"<user-dictionary-package-id>\"]}"
        }
        },
        "analyzer": {
          "sudachi_analyzer": {
            "filter": ["my_searchfilter"],
            "tokenizer": "sudachi_tokenizer",
            "type": "custom"
          }
        },
        "filter":{
          "my_searchfilter": {
            "type": "sudachi_split",
            "mode": "search"
          }
        }
      }
    }
  }
}
```

Actualización de paquetes

En esta sección solo se explica cómo actualizar un paquete de diccionarios personalizados, ya que los paquetes de complementos preempaquetados ya están actualizados. Al cargar una nueva versión de un diccionario en Amazon S3, no se actualiza automáticamente el paquete en Amazon OpenSearch Service. OpenSearch Service almacena su propia copia del archivo, por lo que si carga una nueva versión en S3, debe actualizarla de forma manual.

Cada uno de los dominios asociados también almacena su propia copia del archivo. Para mantener el comportamiento de búsqueda predecible, los dominios siguen utilizando la versión actual del paquete hasta que los actualice explícitamente. Para actualizar un paquete personalizado, modifique el archivo en Amazon S3 Control, actualice el paquete en OpenSearch Service y, a continuación, aplique la actualización.

Consola

1. En la consola de OpenSearch Service, seleccione Paquetes.
2. Seleccione un paquete y luego Actualizar.
3. Proporcione una nueva ruta de S3 al archivo y, a continuación, seleccione Actualizar paquete.
4. Vuelva a la pantalla Paquetes.
5. Cuando el estado del paquete cambie a Disponible, selecciónelo. A continuación, seleccione uno o más dominios asociados, Aplicar actualización y confirme. Espere a que el estado de asociación cambie a Activo.
6. Los siguientes pasos variarán en función de cómo haya configurado los índices:

   - Si el dominio ejecuta OpenSearch o Elasticsearch 7.8 o posterior y solo utiliza analizadores de búsqueda con el campo actualizable establecido en verdadero, no es necesario llevar a cabo ninguna acción adicional. OpenSearch Service actualiza automáticamente los índices mediante la API _plugins/_refresh_search_analyzers.
   - Si el dominio ejecuta Elasticsearch 7.7 o una versión anterior, utiliza analizadores de índice o no utiliza el campo `updateable`, consulte the section called “Actualización manual de los índices con un nuevo diccionario”.

Aunque la consola es el método más simple, también puede utilizar la AWS CLI, los SDK o la API de configuración para actualizar paquetes de OpenSearch Service. Para más información, consulte AWS CLI Command Reference y Amazon OpenSearch Service API Reference.

AWS SDK

En lugar de actualizar de forma manual un paquete en la consola, puede utilizar los SDK para automatizar el proceso de actualización. El siguiente ejemplo de script de Python carga un nuevo archivo de paquete en Amazon S3, actualiza el paquete en OpenSearch Service y aplica el nuevo paquete al dominio especificado. Después de confirmar que la actualización se realizó correctamente, realiza una llamada de ejemplo a OpenSearch para demostrar que los nuevos sinónimos se aplicaron.

Debe proporcionar valores para `host`, `region`, `file_name`, `bucket_name`, `s3_key`, `package_id`, `domain_name` y `query`.

```
from requests_aws4auth import AWS4Auth
import boto3
import requests
import time
import json
import sys

host = ''  # The OpenSearch domain endpoint with https:// and a trailing slash. For
 example, https://my-test-domain.us-east-1.es.amazonaws.com/
region = ''  # For example, us-east-1
file_name = ''  # The path to the file to upload
bucket_name = ''  # The name of the S3 bucket to upload to
s3_key = ''  # The name of the S3 key (file name) to upload to
package_id = ''  # The unique identifier of the OpenSearch package to update
domain_name = ''  # The domain to associate the package with
query = ''  # A test query to confirm the package has been successfully updated

service = 'es'
credentials = boto3.Session().get_credentials()
client = boto3.client('opensearch')
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key,
                   region, service, session_token=credentials.token)


def upload_to_s3(file_name, bucket_name, s3_key):
    """Uploads file to S3"""
    s3 = boto3.client('s3')
    try:
        s3.upload_file(file_name, bucket_name, s3_key)
        print('Upload successful')
        return True
```

```
    except FileNotFoundError:
        sys.exit('File not found. Make sure you specified the correct file path.')


def update_package(package_id, bucket_name, s3_key):
    """Updates the package in OpenSearch Service"""
    print(package_id, bucket_name, s3_key)
    response = client.update_package(
        PackageID=package_id,
        PackageSource={
            'S3BucketName': bucket_name,
            'S3Key': s3_key
        }
    )
    print(response)


def associate_package(package_id, domain_name):
    """Associates the package to the domain"""
    response = client.associate_package(
        PackageID=package_id, DomainName=domain_name)
    print(response)
    print('Associating...')


def wait_for_update(domain_name, package_id):
    """Waits for the package to be updated"""
    response = client.list_packages_for_domain(DomainName=domain_name)
    package_details = response['DomainPackageDetailsList']
    for package in package_details:
        if package['PackageID'] == package_id:
            status = package['DomainPackageStatus']
            if status == 'ACTIVE':
                print('Association successful.')
                return
            elif status == 'ASSOCIATION_FAILED':
                sys.exit('Association failed. Please try again.')
            else:
                time.sleep(10)  # Wait 10 seconds before rechecking the status
                wait_for_update(domain_name, package_id)


def sample_search(query):
    """Makes a sample search call to OpenSearch"""
```

```
path = '_search'
params = {'q': query}
url = host + path
response = requests.get(url, params=params, auth=awsauth)
print('Searching for ' + '"' + query + '"')
print(response.text)
```

> **Note**
>
> Si recibe un error “paquete no encontrado” al ejecutar el script mediante la AWS CLI, es probable que signifique que Boto3 utiliza la región especificada en ~/.aws/config, que no es la región en la que se encuentra el bucket de S3. Puede ejecutar `aws configure` y especificar la región correcta o agregar explícitamente la región al cliente:
>
> ```
> client = boto3.client('opensearch', region_name='us-east-1')
> ```

Actualización manual de los índices con un nuevo diccionario

Las actualizaciones manuales de índices solo se aplican a los diccionarios personalizados, no a los complementos preempaquetados. Para utilizar un diccionario actualizado, debe actualizar de forma manual los índices, si cumple alguna de las siguientes condiciones:

- El dominio ejecuta Elasticsearch 7.7 o anterior.
- Utiliza paquetes personalizados como analizadores de índices.
- Utiliza paquetes personalizados como analizadores de búsqueda, pero no incluye el campo actualizable.

Para actualizar los analizadores con los nuevos archivos de paquete, dispone de dos opciones:

- Cierre y abra los índices que desee actualizar:

```
POST my-index/_close
POST my-index/_open
```

- Vuelva a indexar los índices. Primero, cree un índice que utilice el archivo de sinónimos actualizado (o un archivo completamente nuevo). Tenga en cuenta que solo se admite UTF-8.

```
PUT my-new-index
{
  "settings": {
    "index": {
      "analysis": {
        "analyzer": {
          "synonym_analyzer": {
            "type": "custom",
            "tokenizer": "standard",
            "filter": ["synonym_filter"]
          }
        },
        "filter": {
          "synonym_filter": {
            "type": "synonym",
            "synonyms_path": "analyzers/F222222222"
          }
        }
      }
    }
  },
  "mappings": {
    "properties": {
      "description": {
        "type": "text",
        "analyzer": "synonym_analyzer"
      }
    }
  }
}
```

Luego vuelva a indexar el índice anterior con ese nuevo índice:

```
POST _reindex
{
  "source": {
    "index": "my-index"
  },
  "dest": {
    "index": "my-new-index"
  }
```

```
}
```

Si actualiza con frecuencia los analizadores de índices, utilice alias de índice para mantener una ruta coherente al índice más reciente:

```
POST _aliases
{
  "actions": [
    {
      "remove": {
        "index": "my-index",
        "alias": "latest-index"
      }
    },
    {
      "add": {
        "index": "my-new-index",
        "alias": "latest-index"
      }
    }
  ]
}
```

Si no necesita el índice anterior, elimínelo:

```
DELETE my-index
```

Disociación y eliminación de paquetes

Si disocia un paquete de un dominio, ya sea un diccionario personalizado o un complemento preempaquetado, ya no podrá usar ese paquete cuando cree nuevos índices. Una vez disociado un paquete, los índices existentes que utilizaban el paquete ya no pueden usarlo. Debe eliminar el paquete de cualquier índice antes de poder disociarlo; de lo contrario, la disociación fallará.

La consola es la forma más sencilla de disociar un paquete de un dominio y eliminarlo de OpenSearch Service. La eliminación de un paquete de OpenSearch Service no lo elimina de la ubicación original en Amazon S3.

Disociar un paquete de un dominio

1. Inicie sesión en la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. En el panel de navegación, elija Dominios.
3. Elija el dominio y, a continuación, vaya a la pestaña Paquetes.
4. Seleccione un paquete, Acciones y, a continuación, seleccione Disociar. Confirme su elección.
5. Espere a que el paquete desaparezca de la lista. Puede que tenga que actualizar el navegador.
6. Si desea utilizar el paquete con otros dominios, deténgase aquí. Para continuar con la eliminación del paquete (si es un diccionario personalizado), seleccione Paquetes en el panel de navegación.
7. Seleccione el paquete y elija Eliminar.

Si lo prefiere, puede usar la AWS CLI, los SDK o la API de configuración para disociar y eliminar paquetes. Para más información, consulte AWS CLI Command Reference y Amazon OpenSearch Service API Reference.

Administración de complementos personalizados en Amazon OpenSearch Service

Al utilizar complementos personalizados para OpenSearch Service, puede ampliar la funcionalidad de OpenSearch en áreas como el análisis del idioma, el filtrado personalizado, la clasificación, etc., lo que le permite crear experiencias de búsqueda personalizadas. Se pueden desarrollar complementos personalizados para OpenSearch ampliando la clase `org.opensearch.plugins.Plugin` y luego empaquetándola en un archivo `.zip`.

Amazon OpenSearch Service admite actualmente las siguientes extensiones de complementos:

- AnalysisPlugin: amplía la funcionalidad de análisis al agregar, por ejemplo, analizadores personalizados, tokenizadores de caracteres o filtros para el procesamiento de texto.
- SearchPlugin: mejora las capacidades de búsqueda con tipos de consultas personalizados, algoritmos de similitud, opciones de sugerencias y agregaciones.
- MapperPlugin: le permite crear tipos de campos personalizados y sus configuraciones de mapeo en OpenSearch, lo que le permite definir cómo se deben almacenar e indexar los diferentes tipos de datos.

- ScriptPlugin: le permite agregar capacidades de scripting personalizadas a OpenSearch, por ejemplo, scripts personalizados para operaciones como puntuación, ordenamiento y transformación de valores de campo durante la búsqueda o la indexación.

Puede utilizar la consola de OpenSearch Service o los comandos de API existentes para paquetes personalizados para cargar y asociar el complemento al clúster de Amazon OpenSearch Service. También puede usar el comando DescribePackages para describir todos los paquetes en su cuenta y ver detalles como la versión de OpenSearch y la información de errores. OpenSearch Service valida el paquete de complementos para verificar la compatibilidad de la versión, las vulnerabilidades de seguridad y las operaciones de complementos permitidas. Para obtener más información sobre los paquetes personalizados, consulte the section called “Paquetes”.

Versión de OpenSearch y compatibilidad con la Región de AWS

Los complementos personalizados son compatibles con los dominios de OpenSearch Service que ejecutan OpenSearch versión 2.15 en las siguientes Regiones de AWS:

- Este de EE. UU. (Ohio) (us-east-2)
- Este de EE. UU. (Norte de Virginia) (us-east-1)
- Oeste de EE. UU. (Oregón) (us-west-2)
- Asia-Pacífico (Mumbai) (ap-south-1)
- Asia-Pacífico (Seúl) (ap-northeast-2)
- Asia-Pacífico (Singapur) (ap-southeast-1)
- Asia-Pacífico (Sídney) (ap-southeast-2)
- Asia-Pacífico (Tokio) (ap-northeast-1)
- Canadá (centro) (ca-central-1)
- Europa (Fráncfort) (eu-central-1)
- Europa (Irlanda) (eu-west-1)
- Europa (Londres) (eu-west-2)
- Europa (París) (eu-west-3)
- América del Sur (São Paulo) (sa-east-1)

> **Note**
>
> Los complementos personalizados contienen código desarrollado por el usuario. Cualquier problema, incluido el incumplimiento del SLA, causado por un código desarrollado por el usuario no da derecho a los créditos del SLA. Para obtener más información, consulte el Acuerdo de nivel de Amazon OpenSearch Service.

Temas

- Cuotas de complementos
- Requisitos previos
- Solución de problemas
- Instalación de un complemento personalizado mediante la consola
- Administrar los complementos personalizados mediante la AWS CLI.
- Integración de paquete personalizado de AWS KMS de Amazon OpenSearch Service

Cuotas de complementos

- Puede crear hasta 25 complementos personalizados por región para cada cuenta.
- El tamaño máximo sin comprimir de un complemento es 1 GB.
- El número máximo de complementos que se pueden asociar con un solo dominio es 20. Esta cuota se aplica a todos los tipos de complementos combinados: opcionales, de terceros y personalizados.
- Los complementos personalizados son compatibles con los dominios que ejecuten la versión 2.15 o una posterior de OpenSearch.
- El archivo `descriptor.properties` para su complemento debe ser compatible con una versión de motor similar a 2.15.0 o cualquier versión 2.x.x, donde la versión de parche esté establecida en cero.

Requisitos previos

Antes de instalar un complemento personalizado y asociarlo a un dominio, asegúrese de que cumple los siguientes requisitos:

- La versión de motor compatible con el complemento del archivo `descriptor.properties` debe ser similar a `2.15.0` o `2.x.0`. Es decir, la versión del parche debe ser cero.
- Las siguientes características deben estar habilitadas en su dominio:
  - Cifrado de nodo a nodo
  - Cifrado en reposo
  - `EnforceHTTPS` está establecido en "verdadero"

    Consulte también opensearch-https-required en la Guía para desarrolladores de AWS Config.
  - Los clientes deben admitir Policy-Min-TLS-1-2-PFS-2023-10. Puede especificar esta compatibilidad usando el siguiente comando. Reemplace el *`valor de marcador`* de posición con su propia información:

    ```
    aws opensearch update-domain-config \
        --domain-name domain-name \
        --domain-endpoint-options '{"TLSSecurityPolicy":"Policy-Min-TLS-1-2-
    PFS-2023-10" }'
    ```

    Para obtener más información, consulte DomainEndpointOptions en la Referencia de API de Amazon OpenSearch Service.

Solución de problemas

Si el sistema devuelve el error `PluginValidationFailureReason : The provided plugin could not be loaded`, consulte the section called "La instalación del complemento personalizado falla debido a la compatibilidad de las versiones" para obtener información de solución de problemas.

Instalación de un complemento personalizado mediante la consola

Para asociar un complemento de terceros a un dominio, primero importe la licencia y la configuración del complemento como paquetes.

Para instalar un complemento personalizado

1. Inicie sesión en la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. En el panel de navegación izquierdo, elija Paquetes.
3. Seleccione Importar paquete.

4. En Nombre, ingrese un nombre único y fácil de identificar para el complemento.
5. (Opcional) En la descripción, proporciona cualquier detalle útil sobre el paquete o su propósito.
6. En Tipo de paquete, elija Complemento.
7. En Origen del paquete, ingrese la ruta o busque el archivo ZIP del complemento en Amazon S3.
8. Para la versión del motor OpenSearch, elija la versión de OpenSearch compatible con el complemento.
9. Para el cifrado de paquetes, elija si desea personalizar la clave de cifrado del paquete. De forma predeterminada, OpenSearch Service cifra el paquete de complementos con una Clave propiedad de AWS. En su lugar, puede usar una clave administrada por el cliente.
10. Seleccione Importar.

Después de importar el paquete de complementos, asócielo a un dominio. Para obtener instrucciones, consulte the section called “Para importar y asociar un paquete a un dominio”.

Administrar los complementos personalizados mediante la AWS CLI.

Puede utilizar la AWS CLI para administrar una serie de tareas de complementos personalizados.

Tareas

- Instalación de un complemento personalizado mediante la AWS CLI
- Actualización de un complemento personalizado mediante la AWS CLI.
- Crear o actualizar un complemento personalizado con seguridad de clave AWS KMS
- Actualizar un dominio de OpenSearch Service con complementos personalizados a una versión posterior de OpenSearch mediante la AWS CLI
- Desinstalar y ver el estado de disociación de un complemento personalizado

Instalación de un complemento personalizado mediante la AWS CLI

Antes de empezar

Antes de poder asociar un complemento personalizado con su dominio, debe cargarlo en un bucket de Amazon Simple Storage Service (Amazon S3). El bucket debe estar ubicado en la misma Región de AWS donde usted pretende usar el complemento. Para obtener información sobre cómo agregar un objeto a un bucket de S3, consulte Cargar objetos en la Guía del usuario de Amazon Simple Storage Service.

Si su complemento contiene información sensible, especifique el cifrado del lado del servidor con claves administradas por S3 al cargarlo. Después de cargar el archivo, tome nota de su ruta de S3. El formato de la ruta es `s3://amzn-s3-demo-bucket/file-path/file-name`.

> **Note**
>
> Si lo desea, puede proteger un complemento personalizado al crear el complemento especificando una clave AWS Key Management Service (AWS KMS). Para obtener más información, consulte the section called “Crear o actualizar un complemento personalizado con seguridad de clave AWS KMS”.

Instalación de un complemento personalizado mediante la AWS CLI

1. Cree un nuevo paquete para su complemento personalizado ejecutando el siguiente comando create-package, asegurándose de cumplir con los siguientes requisitos:

   - La ubicación del bucket y la clave debe apuntar al archivo `.zip` del complemento en un bucket de S3 de la cuenta en la que está ejecutando los comandos.
   - El bucket de S3 debe estar en la misma región en la que se está creando el paquete.
   - Solo se admiten archivos `.zip` para los paquetes `ZIP-PLUGIN`.
   - El contenido del archivo `.zip` debe seguir la estructura de directorios esperada por el complemento.
   - El valor de `--engine-version` debe estar en el formato `OpenSearch_{MAJOR}.{MINOR}`. Por ejemplo: **`OpenSearch_2.17`**.

   Reemplace los *valores de marcador de posición* con su información:

   ```
   aws opensearch create-package \
       --package-name package-name \
       --region region \
       --package-type ZIP-PLUGIN \
       --package-source S3BucketName=amzn-s3-demo-bucket,S3Key=s3-key \
       --engine-version opensearch-version
   ```

2. (Opcional) Vea el estado de la operación `create-package`, incluidos los resultados de validación y las vulnerabilidades de seguridad, usando el comando describe-packages. Reemplace los *valores de marcador de posición* con su información:

```
aws opensearch describe-packages \
    --region region  \
    --filters '[{"Name": "PackageType","Value": ["ZIP-PLUGIN"]}, {"Name":
 "PackageName","Value": ["package-name"]}]'
```

El comando devuelve información similar a la siguiente:

```
{
    "PackageDetailsList": [{
        "PackageID": "pkg-identifier",
        "PackageName": "package-name",
        "PackageType": "ZIP-PLUGIN",
        "PackageStatus": "VALIDATION_FAILED",
        "CreatedAt": "2024-11-11T13:07:18.297000-08:00",
        "LastUpdatedAt": "2024-11-11T13:10:13.843000-08:00",
        "ErrorDetails": {
            "ErrorType": "",
            "ErrorMessage": "PluginValidationFailureReason : Dependency Scan
 reported 3 vulnerabilities for the plugin: CVE-2022-23307, CVE-2019-17571,
 CVE-2022-23305"
        },
        "EngineVersion": "OpenSearch_2.15",
        "AllowListedUserList": [],
        "PackageOwner": "OWNER-XXXX"
    }]
}
```

> **Note**
>
> Durante la operación `create-package`, Amazon OpenSearch Service verifica el valor `ZIP-PLUGIN` para la compatibilidad de la versión, las extensiones de complemento compatibles y las vulnerabilidades de seguridad. Las vulnerabilidades de seguridad se escanean mediante el servicio Amazon Inspector. Los resultados de estas verificaciones se muestran en el campo `ErrorDetails` de la respuesta de la API.

3. Utilice el comando associate-package para asociar el complemento al dominio de OpenSearch Service que elija utilizando el ID de paquete del paquete creado en el paso anterior.

> **Tip**
>
> Si tiene varios complementos, puede usar el comando associate-packages para asociar múltiples paquetes a un dominio en una sola operación.

Reemplace los *valores de marcador de posición* con su información:

```
aws opensearch associate-package \
    --domain-name domain-name \
    --region region \
    --package-id package-id
```

> **Note**
>
> El complemento se instala y desinstala utilizando un proceso de implementación azul/verde.

4. (Opcional) Use el comando list-packages-for-domain para ver el estado de la asociación. El estado de la asociación cambia a medida que el flujo de trabajo avanza de `ASSOCIATING` a `ACTIVE`. El estado de la asociación cambia a ACTIVE cuando se completa la instalación del complemento y el complemento está listo para su uso.

   Reemplace los *valores de marcador de posición* con su información.

```
aws opensearch list-packages-for-domain \
    --region region \
    --domain-name domain-name
```

Actualización de un complemento personalizado mediante la AWS CLI.

Use el comando update-package para realizar cambios en un complemento.

> **Note**
>
> Si lo desea, puede proteger un complemento personalizado al actualizar el complemento especificando una clave AWS Key Management Service (AWS KMS). Para obtener más

información, consulte the section called “Crear o actualizar un complemento personalizado con seguridad de clave AWS KMS”.

Para actualizar un complemento personalizado mediante la AWS CLI

- Ejecute el siguiente comando. Reemplace los *valores de marcador de posición* con su información.

```
aws opensearch update-package \
    --region region \
    --package-id package-id \
    --package-source S3BucketName=amzn-s3-demo-bucket,S3Key=s3-key \
    --package-description description
```

Después de actualizar un paquete, puede usar el comando associate-package o associate-packages para aplicar las actualizaciones del paquete a un dominio.

> **Note**
>
> Puede auditar, crear, actualizar, asociar y desasociar operaciones en el complemento mediante AWS CloudTrail. Para obtener más información, consulte Monitoreo de las llamadas a la API de Amazon OpenSearch Service con AWS CloudTrail.

Crear o actualizar un complemento personalizado con seguridad de clave AWS KMS

Puede proteger un complemento personalizado al crearlo o actualizarlo especificando una clave AWS KMS. Para lograr esto, configure `PackageEncryptionOptions` en `true` y especifique el Nombre de recurso de Amazon (ARN) de la clave, como se muestra en los siguientes ejemplos.

Ejemplo: cree un complemento personalizado con una clave de seguridad de AWS KMS

```
aws opensearch create-package \
    --region us-east-2  --package-name my-custom-package \
    --package-type ZIP-PLUGIN \
    --package-source S3BucketName=amzn-s3-demo-bucket,S3Key=my-s3-key
    --engine-version OpenSearch_2.15
"PackageConfigOptions": {
```

```
     ...
  }
  "PackageEncryptionOptions": {
    "Enabled": true,
    "KmsKeyId":"arn:aws:kms:us-east-2:111222333444:key/2ba228d5-1d09-456c-ash9-
daf42EXAMPLE"
  }
```

Ejemplo: actualice un complemento personalizado con una clave de seguridad de AWS KMS

```
aws opensearch update-package \
    --region us-east-2  --package-name my-custom-package \
    --package-type ZIP-PLUGIN \
    --package-source S3BucketName=amzn-s3-demo-bucket,S3Key=my-s3-key
    --engine-version OpenSearch_2.15
"PackageConfigOptions": {
     ...
  }
  "PackageEncryptionOptions": {
    "Enabled": true,
    "KmsKeyId":"arn:aws:kms:us-east-2:111222333444:key/2ba228d5-1d09-456c-ash9-
daf42EXAMPLE"
  }
```

> **Important**
>
> Si la clave AWS KMS que usted especifica está deshabilitada o eliminada, puede dejar el clúster asociado inoperativo.

Para obtener más información sobre la integración de AWS KMS con paquetes personalizados, the section called “Integración de paquete personalizado de AWS KMS de Amazon OpenSearch Service”.

Actualizar un dominio de OpenSearch Service con complementos personalizados a una versión posterior de OpenSearch mediante la AWS CLI

Cuando necesite actualizar un dominio del servicio OpenSearch que utiliza complementos personalizados a una versión posterior de OpenSearch, complete los siguientes procesos.

Para actualizar un dominio de OpenSearch Service con complementos personalizados a una versión posterior de OpenSearch mediante la AWS CLI.

1. Use el comando create-package para crear un nuevo paquete para su complemento especificando la nueva versión de OpenSearch.

   Asegúrese de que el nombre del paquete sea el mismo para el complemento en todas las versiones del motor. Si se cambia el nombre del paquete, se produce un error en el proceso de actualización del dominio durante la implementación azul/verde.
2. Actualice su dominio a una versión superior siguiendo los pasos que se indican en the section called “Actualización de dominios”.

   Durante este proceso, Amazon OpenSearch Service desasocia la versión anterior del paquete de complementos e instala la nueva versión mediante una implementación azul/verde.

Desinstalar y ver el estado de disociación de un complemento personalizado

Para desinstalar el complemento de cualquier dominio, puede usar el comando dissociate-package. Al ejecutar este comando, también se eliminan todos los paquetes de configuración o licencias relacionados. A continuación, puede utilizar el comando list-packages-for-domain para ver el estado de la disociación.

> **Tip**
>
> También puede usar el comando dissociate-packages para desinstalar varios complementos de un dominio en una sola operación.

Desinstalar y ver el estado de disociación de un complemento personalizado

1. Deshabilita el complemento en todos los índices. Esto debe hacerse antes de disociar el paquete de complementos.

   Si intenta desinstalar un complemento antes de deshabilitarlo en cada índice, el proceso de implementación azul/verde permanecerá atascado en el estado `Processing`.
2. Ejecute el siguiente comando para desinstalar el complemento. Reemplace los *`valores de marcador de posición`* con su información.

```
aws opensearch dissociate-package \
```

```
    --region region \
    --package-id plugin-package-id \
    --domain-name domain name
```

3. (Opcional) Ejecute el comando list-packages-for-domain para ver el estado de la disociación.

Integración de paquete personalizado de AWS KMS de Amazon OpenSearch Service

Los paquetes personalizados de Amazon OpenSearch Service proporcionan cifrado por defecto para proteger sus paquetes `ZIP-PLUGIN` en reposo utilizando Claves administradas por AWS.

- Claves propiedad de AWS: los paquetes personalizados de Amazon OpenSearch Service utilizan estas claves de forma predeterminada para cifrar automáticamente sus paquetes `ZIP-PLUGIN`. No puede ver, administrar ni usar Claves propiedad de AWS ni auditar su uso. Sin embargo, no necesita realizar ninguna acción ni cambiar ningún programa para proteger las claves que cifran los datos. Para obtener más información, consulte Claves propiedad de AWS en la Guía para desarrolladores de AWS Key Management Service.
- Claves administradas por el cliente: puede agregar una segunda capa de cifrado sobre las Claves propiedad de AWS existentes al elegir una clave administrada por el cliente al crear su paquete personalizado `ZIP-PLUGIN`.

  Los paquetes personalizados de Amazon OpenSearch Service admiten el uso de una clave simétrica administrada por el cliente que usted crea, posee y administra para agregar una segunda capa de cifrado sobre el cifrado administrado por AWS existente. Debido a que usted tiene control total sobre esta capa de cifrado, puede realizar las siguientes tareas:

  - Establecer y mantener políticas de claves.
  - Establecer y mantener políticas y permisos de AWS Identity and Access Management (IAM)
  - Habilitar y deshabilitar políticas de claves.
  - Rotar el material criptográfico.
  - Agregar etiquetas.
  - Crear alias de claves.
  - Programar la eliminación de claves.

Para más información, consulte las claves administradas por el cliente en la Guía para desarrolladores de AWS Key Management Service.

> **Note**
>
> Los paquetes personalizados de Amazon OpenSearch Service habilitan automáticamente el cifrado en reposo utilizando Claves propiedad de AWS sin cargo. Sin embargo, se aplican cargos de AWS KMS cuando utiliza una clave administrada por el cliente. Para obtener más información acerca de los precios, consulte Precios de AWS Key Management Service.

Cómo el servicio de paquetes personalizados de Amazon OpenSearch Service utiliza permisos en AWS KMS

Los paquetes personalizados de OpenSearch Service requieren una concesión para utilizar su clave administrada por el cliente.

Cuando crea un paquete `ZIP-PLUGIN` cifrado con una clave administrada por el cliente, el servicio de paquetes personalizados de Amazon OpenSearch Service crea un permiso en su nombre enviando una solicitud CreateGrant a AWS KMS. Los permisos en AWS KMS otorgan a OpenSearch Service acceso a una clave AWS KMS en su cuenta. Los permisos creados por los paquetes personalizados de OpenSearch Service tienen una restricción que permite operaciones únicamente cuando la solicitud incluye un contexto de cifrado con el ID de su paquete personalizado.

Los paquetes personalizados de Amazon OpenSearch Service requieren el permiso para usar su clave administrada por el cliente en las siguientes operaciones internas:

Operación	Descripción
`DescribeKey`	Enviar solicitudes `DescribeKey` a AWS KMS para verificar que el ID de clave simétrica administrada por el cliente ingresado al crear el paquete del complemento sea válido.
`GenerateDataKeyWithoutPlaintext`	Envía solicitudes `GenerateDataKeyWithoutPlaintext` a AWS KMS para generar claves de datos cifradas con su clave administrada por el cliente.
`GenerateDataKey`	Envía solicitudes `GenerateDataKey` a AWS KMS para generar claves de datos para cifrar el paquete al copiarlo internamente.
`Decrypt`	Envía solicitudes `Decrypt` a AWS KMS para descifrar las claves de datos cifradas para que puedan usarse para descifrar sus datos.

Puede revocar el acceso a la concesión o eliminar el acceso del servicio a la clave administrada por el cliente en cualquier momento. Si no lo hace, OpenSearch Service no podrá acceder a ningún dato cifrado con la clave administrada por el cliente, lo que afecta a las operaciones que dependen de esos datos. Por ejemplo, si intenta asociar un paquete de complemento al que OpenSearch Service no puede acceder, la operación devuelve un error `AccessDeniedException`.

Crear una clave administrada por el cliente

Puede crear una clave administrada por el cliente a través de la Consola de administración de AWS o la API de AWS KMS.

Para crear una clave simétrica administrada por el cliente

- Siga los pasos en Crear una clave KMS en la Guía para desarrolladores de AWS Key Management Service.

Política de claves

Las políticas de clave controlan el acceso a la clave administrada por el cliente. Cada clave administrada por el cliente debe tener exactamente una política de clave, que contiene instrucciones que determinan quién puede usar la clave y cómo puede utilizarla. Cuando crea la clave administrada por el cliente, puede especificar una política de clave. Para obtener más información, consulte Políticas de claves en AWS KMS en la Guía para desarrolladores de AWS Key Management Service.

Para usar su clave administrada por el cliente con los recursos de su complemento, debe permitir las siguientes operaciones de API en la política de claves:

- `kms:CreateGrant`: agrega una concesión a una clave administrada por el cliente. Permite el control del acceso a una clave AWS KMS especificada, permitiendo el acceso a las operaciones de permisos que requieren los paquetes personalizados de OpenSearch Service. Para obtener más información sobre el uso de permisos, consulte la Guía para desarrolladores de AWS KMS.

  Esto permite que OpenSearch Service realice lo siguiente:

  - Llamar a `GenerateDataKeyWithoutPlainText` para generar una clave de datos cifrada y almacenarla para validaciones posteriores.
  - Llamar a `GenerateDataKey` para copiar internamente el paquete del complemento.
  - Llamar a `Decrypt` para acceder internamente al paquete del complemento.

  - Configurar una entidad principal que se retire para permitir que el servicio `RetireGrant`.
- `kms:DescribeKey`: proporciona los detalles de la clave administrada por el cliente para permitir que OpenSearch Service valide la clave.
- `kms:GenerateDataKey`, `kms:GenerateDataKeyWithoutPlaintext`, `kms:Decrypt` otorgan a los paquetes personalizados de OpenSearch Service acceso para usar estas operaciones en el permiso.

Los siguientes son ejemplos de declaraciones de política que puede agregar para los paquetes personalizados de OpenSearch Service:

```
"Statement" : [
  {
    "Sid" : "Allow access to principals authorized to use OpenSearch Service custom
 packages",
    "Effect" : "Allow",
    "Principal" : {
      "AWS" : "*"
    },
    "Action" : [
      "kms:CreateGrant",
      "kms:GenerateDataKey",
      "kms:GenerateDataKeyWithoutPlaintext",
      "kms:Decrypt"
    ],
    "Resource" : "*",
    "Condition" : {
      "StringEquals" : {
        "kms:ViaService" : "custom-packages.region.amazonaws.com"
      },
      "StringEquals" : {
        "kms:EncryptionContext:packageId": "Id of the package"
      }
    }
  },
  {
    "Sid" : "Allow access to principals authorized to use Amazon OpenSearch Service
 custom packages",
    "Effect" : "Allow",
    "Principal" : {
      "AWS" : "*"
```

```
    },
    "Action" : [
      "kms:DescribeKey"
    ],
    "Resource" : "*",
    "Condition" : {
      "StringEquals" : {
        "kms:ViaService" : "custom-packages.region.amazonaws.com"
      }
    }
  }
]
```

Para obtener más información sobre cómo especificar permisos en una política, consulte Políticas de clave en AWS KMS y en la Guía para desarrolladores de AWS Key Management Service.

Para obtener más información sobre cómo solucionar problemas de acceso a claves, consulte Solución de problemas de permisos de AWS KMS en la Guía para desarrolladores de AWS Key Management Service.

Especificar una clave administrada por el cliente para los paquetes personalizados de Amazon OpenSearch Service

Puede especificar una clave administrada por el cliente como segunda capa de cifrado para sus paquetes `ZIP-PLUGIN`.

Al crear un paquete de complemento, puede especificar la clave de datos ingresando un ID de clave AWS KMS, que los paquetes personalizados de OpenSearch Service usan para cifrar el paquete del complemento.

ID de clave AWS KMS: un identificador para una clave administrada por el cliente AWS KMS. Introduzca el ID de la clave, el ARN de la clave, el nombre de alias o el ARN del alias.

Contexto de cifrado de los paquetes personalizados de Amazon OpenSearch Service

Un contexto de cifrado es un conjunto opcional de pares clave-valor que pueden contener información contextual adicional sobre los datos.

AWS KMS utiliza el contexto de cifrado como datos autenticados adicionales para admitir el cifrado autenticado. Cuando se incluye un contexto de cifrado en una solicitud para cifrar datos, AWS KMS

vincula el contexto de cifrado a los datos cifrados. Para descifrar los datos, debe incluir el mismo contexto de cifrado en la solicitud.

Contexto de cifrado de los paquetes personalizados de Amazon OpenSearch Service

Los paquetes personalizados de Amazon OpenSearch Service utilizan el mismo contexto de cifrado en todas las operaciones criptográficas AWS KMS, donde la clave es `packageId` y el valor es el `package-id` de su paquete de complemento.

Uso del contexto de cifrado para monitoreo

Cuando utiliza una clave simétrica administrada por el cliente para cifrar su paquete de complemento, puede usar el contexto de cifrado en los registros de auditoría y logs para identificar cómo se está utilizando la clave administrada por el cliente. El contexto de cifrado también aparece en los registros generados por AWS CloudTrail o registros de Amazon CloudWatch.

Utilizar el contexto de cifrado para controlar el acceso a la clave administrada por el cliente

Puede utilizar el contexto de cifrado en políticas de claves y políticas de IAM como condiciones para controlar el acceso a su clave simétrica administrada por el cliente. Puede usar también una restricción de contexto de cifrado en una concesión.

Los paquetes personalizados de OpenSearch Service utilizan una restricción de contexto de cifrado en los permisos para controlar el acceso a la clave administrada por el cliente en su cuenta o región. La restricción de concesión requiere que las operaciones que permite la concesión utilicen el contexto de cifrado especificado.

Los siguientes son ejemplos de declaraciones de política de claves para conceder acceso a una clave administrada por el cliente para un contexto de cifrado específico. La condición de esta declaración de política exige que las concesiones tengan una restricción de contexto de cifrado que especifique el contexto de cifrado.

```
{
    "Sid": "Enable DescribeKey",
    "Effect": "Allow",
    "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/ExampleReadOnlyRole"
    },
    "Action": "kms:DescribeKey",
    "Resource": "*"
},
```

```
{
    "Sid": "Enable OpenSearch Service custom packages to use the key",
    "Effect": "Allow",
    "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/ExampleReadOnlyRole"
    },
    "Action" : [
         "kms:CreateGrant",
        "kms:GenerateDataKey",
        "kms:GenerateDataKeyWithoutPlaintext",
        "kms:Decrypt"
    ],
    "Resource": "*",
    "Condition": {
        "StringEquals" : {
            "kms:EncryptionContext:packageId": "ID of the package"
         }
    }
}
```

Monitoreo de sus claves de cifrado para el servicio de paquetes personalizados de OpenSearch

Cuando utiliza una clave administrada por el cliente AWS KMS con los recursos del servicio de paquetes personalizados de OpenSearch Service, puede usar CloudTrail o Registros de CloudWatch para rastrear las solicitudes que los paquetes personalizados de OpenSearch envían a AWS KMS.

Más información

Los siguientes recursos proporcionan más información sobre cifrado de datos en reposo.

- Para obtener más información sobre los conceptos básicos de AWS KMS, consulte AWS KMS keys en la Guía para desarrolladores de AWS Key Management Service.
- Para obtener más información sobre las prácticas recomendadas de seguridad para AWS KMS, consulte la guía Prescriptive Guidance de AWS para las prácticas recomendadas de AWS Key Management Service.

Instalación de complementos de terceros en Amazon OpenSearch Service

Amazon OpenSearch Service admite complementos de terceros de socios seleccionados. Estos complementos pueden mejorar su configuración de OpenSearch con características adicionales,

como analizadores personalizados, tokenizadores o capacidades de cifrado. Siga las instrucciones específicas de instalación y configuración proporcionadas por los desarrolladores de terceros para garantizar una integración adecuada con su dominio de OpenSearch Service.

> **Note**
>
> Debe obtener y mantener licencias válidas directamente de los desarrolladores de terceros. Algunos proveedores podrían no habilitar sus complementos en todas las Regiones de AWS, por lo que debe consultar con el proveedor del complemento sobre su disponibilidad.

Los siguientes complementos de terceros están disponibles para su uso con OpenSearch Service:

- Complemento de cifrado Portal26 (Titanium-Lockbox): Utiliza cifrado certificado NIST FIPS 140-2 para cifrar datos mientras se indexan. Incluye soporte Bring Your Own Key (BYOK), que le permite administrar sus claves de cifrado para mayor seguridad. Este complemento es proporcionado por Portal26 y requiere OpenSearch versión 2.15 o superior.
- Concordancia de nombres (RNI): Coincide nombres, organizaciones, direcciones y fechas en más de 24 idiomas, lo que mejora la seguridad y el cumplimiento normativo. Este complemento es proporcionado por Babel Street y requiere OpenSearch versión 2.15 o superior.

Temas

- Requisitos previos
- Instalación de complementos de terceros
- Pasos a seguir a continuación

Requisitos previos

Antes de instalar un complemento de terceros, realice los siguientes pasos:

- Obtenga los archivos de configuración y licencia del complemento y cárguelos en un bucket de Amazon S3. El bucket debe estar en la misma Región de AWS que el dominio.
- Un complemento de terceros es un tipo de complemento personalizado. Asegúrese de que el dominio cumpla con los requisitos previos para complementos personalizados.

Instalación de complementos de terceros

Para asociar un complemento de terceros con un dominio de OpenSearch Service, primero debe cargar tres paquetes separados: el paquete de licencia, el paquete de configuración y el paquete del complemento.

- Paquete de licencia: incluye la información de licencia o metadatos asociados con el complemento, en formato .json o .xml.
- Paquete de configuración: contiene los archivos de configuración del complemento, los recursos de soporte y la configuración. Estos archivos definen cómo se comporta o se integra el complemento con OpenSearch.
- Paquete del complemento: contiene el binario compilado del complemento, que es el código ejecutable que OpenSearch ejecuta. Este es el núcleo de la funcionalidad del complemento.

Después de cargar los paquetes, puede asociar el complemento y la licencia con un dominio compatible.

Consola

Para asociar un complemento de terceros a un dominio, primero importe la licencia y la configuración del complemento como paquetes.

Para instalar un complemento de terceros

1. Inicie sesión en la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. En el panel de navegación izquierdo, elija Paquetes.
3. En primer lugar, importe el paquete de licencias. Seleccione Importar paquete.
4. En Tipo de paquete, elija Licencia.
5. En Origen del paquete, ingrese la ruta al archivo JSON o XML de la licencia en Amazon S3.
6. Seleccione Importar. El paquete aparece en la pestaña Licencias de la página Paquetes.
7. Ahora, importe la configuración del complemento. Seleccione nuevamente Importar paquete.
8. En Tipo de paquete, elija Configuración.
9. Para Origen del paquete, ingrese la ruta al archivo ZIP de la configuración del complemento en Amazon S3.
10. Seleccione Importar.

11. Por último, importe el propio complemento. Seleccione Importar paquete.
12. En Tipo de paquete, elija Complemento.
13. En Origen del paquete, ingrese la ruta del archivo ZIP del complemento en Amazon S3.
14. Seleccione la versión del motor de búsqueda OpenSearch compatible con el complemento.
15. Seleccione Importar.

Para asociar un complemento de terceros a un dominio

1. Ahora, asocie la licencia y la configuración del complemento al dominio. En el panel de navegación izquierdo, seleccione Dominios.
2. Elija el nombre del dominio para abrir su configuración de clúster.
3. Vaya a la pestaña Complementos.
4. Elija Asociar paquetes y seleccione los paquetes de complementos, licencias y configuración que acaba de importar.
5. Elija Seleccionar.
6. Elija Siguiente. Revise los paquetes que desea asociar y elija Asociar.

CLI

Primero, use el comando create-package para crear un paquete nuevo que contenga la licencia del complemento. La `S3Key` debe apuntar a un archivo .json o .xml en Amazon S3 que incluya el texto de la licencia o los metadatos.

```
aws opensearch create-package \
  --package-name plugin-license-package \
  --package-type PACKAGE-LICENSE \
  --package-source S3BucketName=my-bucket,S3Key=licenses/my-plugin-license.json
```

Use el comando create-package nuevamente para crear un paquete que contenga la configuración del complemento. La `S3Key` debe apuntar a un archivo .zip en Amazon S3 que cumpla con la estructura de directorios esperada por el complemento.

```
aws opensearch create-package \
  --package-name plugin-config-package \
  --package-type PACKAGE-CONFIG \
  --package-source S3BucketName=my-bucket,S3Key=path/to/package.zip
```

Use nuevamente el comando create-package para crear un paquete que contenga el complemento en sí. La S3Key debe apuntar al archivo .zip del complemento en Amazon S3.

```
aws opensearch create-package \
  --package-name plugin-package \
  --package-type ZIP-PLUGIN \
  --package-source S3BucketName=my-bucket,S3Key=path/to/package.zip
```

Finalmente, use el comando associate-package para vincular el complemento del socio, la licencia y la configuración a un dominio compatible, especificando los ID de paquete de cada uno. Especifique el ID del complemento como un requisito previo para los demás paquetes; esto significa que debe asociarse al dominio antes que los otros paquetes.

```
aws opensearch associate-packages \
  --domain-name my-domain \
  --package-list '[{"PackageID": "plugin-package-id"},{"PackageID": "license-package-
id","PrerequisitePackageIDList":["plugin-package-id"]},{"PackageID":"config-package-
id","PrerequisitePackageIDList":["plugin-package-id"]}]'
```

Pasos a seguir a continuación

Cuando la asociación se complete, usted podrá habilitar el complemento en índices específicos o configurarlo según sus necesidades. Para aplicar la funcionalidad del complemento de terceros a índices específicos, modifique la configuración del índice durante su creación o actualice índices existentes. Por ejemplo, si su complemento de terceros incluye un analizador personalizado, haga referencia a este en la configuración del índice.

Para aplicar las características del complemento de manera uniforme en varios índices, use plantillas de índice que incluyan las configuraciones del complemento. Consulte siempre la documentación del complemento para comprender cómo configurar sus características en su entorno de OpenSearch.

Consultar los datos de Amazon OpenSearch Service con SQL

Puede utilizar SQL para consultar su Amazon OpenSearch Service, en lugar de utilizar la Consulta de OpenSearch DSL con base en JSON. Consultar con SQL es útil si ya está familiarizado con el lenguaje o si desea integrar su dominio en una aplicación que lo utilice. La compatibilidad con SQL está disponible en los dominios que ejecutan OpenSearch o Elasticsearch 6.5 o versiones posteriores.

> **Note**
>
> Esta documentación describe la compatibilidad de versiones entre OpenSearch Service y varias versiones del complemento SQL, así como los controladores JDBC y ODBC. Consulte la documentación de código abierto de OpenSearch para obtener información sobre la sintaxis de las consultas básicas y complejas, las funciones, las consultas de metadatos y las funciones agregadas.

Utilice la siguiente tabla para buscar la versión del complemento SQL compatible con cada versión de OpenSearch y Elasticsearch.

OpenSearch

Versión de OpenSearch	Versión del complemento de SQL	Características notables
2.19.0	2.19.0.0	
2.18.0	2.18.0.0	
2.17.0	2.17.0.0	
2.15.0	2.15.0.0	
2.13.0	2.13.0.0	
2.11.0	2.11.0.0	Adición de compatibilidad con el lenguaje y las consultas de PPL
2.9.0	2.9.0.0	Se añadió el conector Spark y admite funciones de tabla y ProMQL
2.7.0	2.7.0.0	Se añadió API `datasource`
2.5.0	2.5.0.0	
2.3.0	2.3.0.0	Se añadieron funciones de fecha y hora `maketime` y `makedate`

Versión de OpenSearch	Versión del complemento de SQL	Características notables
1.3.0	1.3.0.0	Admite tamaño límite de consulta predeterminado y cláusula IN al seleccionar de una lista de valores
1.2.0	1.2.0.0	Se añadió nuevo protocolo para el formato de respuesta de visualización
1.1.0	1.1.0.0	Admite la función de coincidencia como filtro en SQL y PPL
1.0.0	1.0.0.0	Admite la consulta de un flujo de datos

Open Distro para Elasticsearch

Versión de Elasticsearch	Versión del complemento de SQL	Características notables
7.10	1.13.0	NULL FIRST y LAST para funciones de ventana, función CAST(), comandos SHOW y DESCRIBE
7.9	1.11.0	Se añadieron funciones de fecha/hora adicionales, palabra clave ORDER BY
7.8	1.9.0	
7.7	1.8.0	
7.3	1.3.0	Varios operadores de cadena y número
7.1	1.1.0	

Ejemplo de llamada

Para consultar los datos con SQL, envíe solicitudes HTTP a `_sql` con el formato siguiente:

```
POST domain-endpoint/_plugins/_sql
{
  "query": "SELECT * FROM my-index LIMIT 50"
}
```

> **Note**
>
> Si el dominio ejecuta Elasticsearch en lugar de OpenSearch, el formato es `_opendistro/_sql`.

Notas y diferencias

Las llamadas a `_plugins/_sql` incluyen los nombres de índice en el cuerpo de la solicitud y, por lo tanto, se aplican las mismas consideraciones de política de acceso que en las operaciones bulk, mget y msearch. Como siempre, siga el principio de privilegios mínimos cuando conceda permisos a las operaciones de API.

Para consideraciones de seguridad relacionadas con la utilización de SQL con el control de acceso detallado, consulte the section called “Control de acceso detallado”.

El complemento SQL de OpenSearch incluye muchas opciones de configuración ajustables. En OpenSearch Service, utilice la ruta `_cluster/settings`, en lugar de la ruta de configuración del complemento (`_plugins/_query/settings`):

```
PUT _cluster/settings
{
  "transient" : {
    "plugins.sql.enabled" : true
  }
}
```

En los dominios de Elasticsearch heredados, reemplace `plugins` con `opendistro`:

```
PUT _cluster/settings
{
  "transient" : {
    "opendistro.sql.enabled" : true
  }
```

```
}
```

SQL Workbench

SQL Workbench es una interfaz de usuario de OpenSearch Dashboards que permite ejecutar consultas SQL bajo demanda, traducir SQL en su equivalente REST y ver y guardar resultados como texto, JSON, JDBC o CSV. Para más información, consulte Query Workbench.

CLI SQL

La CLI SQL es una aplicación Python independiente que se puede iniciar con el comando `opensearchsql`. Para obtener información sobre los pasos para instalar, configurar y utilizar, consulte SQL CLI.

Controlador JDBC

El controlador de Java Database Connectivity (JDBC) permite integrar dominios de OpenSearch Service en sus aplicaciones favoritas de inteligencia empresarial (BI). Para descargar el controlador, haga clic aquí. Para más información, consulte el repositorio de GitHub.

Controlador ODBC

El controlador de Open Database Connectivity (ODBC) es un controlador de ODBC de solo lectura para Windows y macOS que permite conectar aplicaciones de inteligencia empresarial y visualización de datos como Microsoft Excel al complemento SQL.

Para obtener más información sobre cómo instalar el controlador, consulte el Repositorio SQL en GitHub.

Búsqueda en clústeres de Amazon OpenSearch Service

La búsqueda entre clústeres en Amazon OpenSearch Service permite realizar consultas y agregaciones en varios dominios conectados. A menudo tiene más sentido usar varios dominios más pequeños en lugar de un solo dominio grande, especialmente cuando se ejecutan distintos tipos de cargas de trabajo.

Los dominios específicos de la carga de trabajo permiten realizar las siguientes tareas:

- Optimice cada dominio mediante la elección de tipos de instancia para cargas de trabajo específicas.

- Establezca límites de aislamiento de errores entre cargas de trabajo. Esto significa que si una de las cargas de trabajo devuelve un error, el error queda contenido dentro de ese dominio específico y no afecta a las demás cargas de trabajo.
- Escale más fácilmente entre dominios.

La búsqueda entre clústeres admite OpenSearch Dashboards, por lo que puede crear visualizaciones y paneles en todos los dominios. Paga los cargos de estándar de transferencia de datos de AWS para los resultados de búsqueda transferidos entre dominios.

> **Note**
>
> OpenSearch de código abierto también tiene documentación para la búsqueda entre clústeres. La configuración difiere considerablemente entre los clústeres de código abierto en comparación con los dominios administrados de Amazon OpenSearch Service. En concreto, en OpenSearch Service, las conexiones entre clústeres se configuran mediante la Consola de administración de AWS en lugar de a través de cURL. Además, el servicio administrado utiliza AWS Identity and Access Management (IAM) para la autenticación entre clústeres, además de un control de acceso detallado. Por lo tanto, recomendamos utilizar esta documentación, en lugar de la documentación de OpenSearch de código abierto, para configurar la búsqueda entre clústeres para sus dominios.

Temas

- Limitaciones
- Requisitos previos de búsqueda entre clústeres
- Precio de búsqueda entre clústeres
- Configuración de una conexión
- Eliminación de una conexión
- Configuración de seguridad y explicación de ejemplo
- OpenSearch Dashboards

Limitaciones

La búsqueda en clústeres tiene varias limitaciones importantes:

- No puede conectar un dominio de Elasticsearch a un dominio de OpenSearch.
- No puede conectarse a clústeres de OpenSearch o Elasticsearch autoadministrados.
- Para conectar dominios de distintas regiones, ambos dominios deben estar en Elasticsearch 7.10 o posterior u OpenSearch.
- Un dominio puede tener un máximo de 20 conexiones salientes. Del mismo modo, un dominio puede tener un máximo de 20 conexiones entrantes. En otras palabras, un dominio puede conectarse a un máximo de 20 dominios.
- El dominio de origen debe estar en la misma versión o superior a la del dominio de destino. Si configura una conexión bidireccional entre dos dominios y desea actualizar uno o ambos, antes debe eliminar una de las conexiones.
- No puede usar diccionarios personalizados o SQL con la búsqueda en clústeres.
- No puede usar CloudFormation para conectar dominios.
- No se puede utilizar la búsqueda entre clústeres en instancias M3 o bursátiles (T2 y T3).

Requisitos previos de búsqueda entre clústeres

Antes de configurar la búsqueda en clústeres, asegúrese de que los dominios cumplan los siguientes requisitos:

- Dos dominios de OpenSearch, o dominios de Elasticsearch en la versión 6.7 o posterior
- Control de acceso detallado habilitado
- Cifrado de nodo a nodo habilitado

Precio de búsqueda entre clústeres

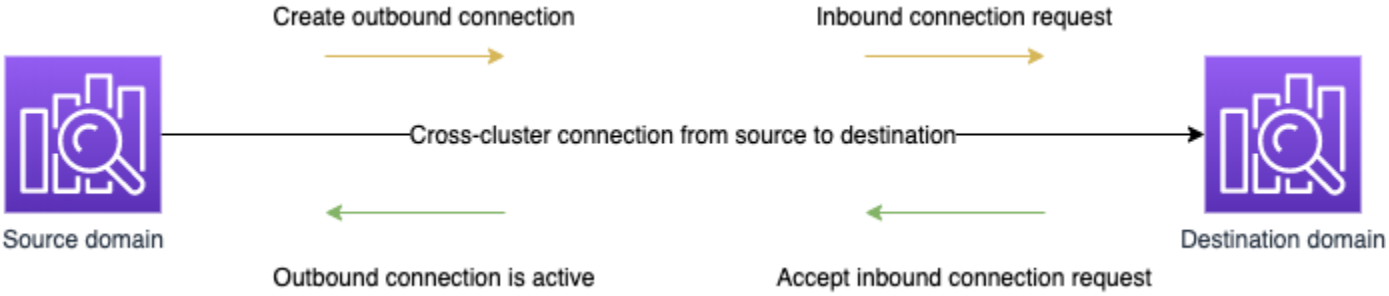
No hay ningún cargo adicional por buscar entre dominios.

Configuración de una conexión

El dominio de “origen” hace referencia al dominio desde el que se origina una petición de búsqueda en clústeres. En otras palabras, el dominio de origen es al que envía la petición de búsqueda inicial.

El dominio “destino” es el dominio que consulta el dominio de origen.

Una conexión entre clústeres es unidireccional desde el dominio de origen hasta el dominio de destino. Esto significa que el dominio de destino no puede consultar el dominio de origen. Sin embargo, puede configurar otra conexión en la dirección opuesta.



El dominio de origen crea una conexión “saliente” al dominio de destino. El dominio de destino recibe una solicitud de conexión “entrante” del dominio de origen.

Para configurar una conexión

1. En el panel del dominio, seleccione un dominio y vaya a la pestaña Conexiones.
2. En la sección Conexiones de salida, seleccione Solicitar.
3. En Alias de conexión, ingrese un nombre para la conexión.
4. Seleccione entre conectarse a un dominio de su Cuenta de AWS y región, o hacerlo a otra cuenta o región.

   - Para conectarse a un clúster en la Cuenta de AWS y la región, seleccione el dominio en el menú desplegable y luego Solicitar.
   - Para conectarse a un clúster de otra Cuenta de AWS o región, seleccione el ARN del dominio remoto y elija Solicitar. Para conectar dominios en distintas regiones, ambos dominios deben ejecutar Elasticsearch versión 7.10 o posterior u OpenSearch.
5. Para omitir los clústeres no disponibles para las consultas de clústeres, seleccione Omitir los no disponibles. Esta configuración garantiza que las consultas entre clústeres devuelvan resultados parciales a pesar de que se produzcan errores en uno o más clústeres remotos.
6. La búsqueda entre clústeres valida primero la solicitud de conexión para asegurarse de que se cumplen los requisitos previos. Si los dominios son incompatibles, la solicitud de conexión entra en el estado `Validation failed`.
7. Una vez correctamente validada la solicitud de conexión, se envía al dominio de destino, donde tiene que aprobarse. Mientras no se obtenga la aprobación, la conexión permanecerá en el estado de `Pending acceptance`. Cuando se acepta la solicitud de conexión en el dominio de destino, el estado cambia a `Active` y el dominio de destino se vuelve disponible para consultas.

- La página de dominio muestra el estado general del dominio y los detalles del estado de la instancia del dominio de destino. Los propietarios de dominios tienen la flexibilidad de crear, visualizar, eliminar y monitorear conexiones desde o hacia sus dominios.

Una vez establecida la conexión, se cifra todo el tráfico que circule entre los nodos de los dominios conectados. Si conecta un dominio de VPC a un dominio que no es VPC y el dominio que no es VPC es un punto de conexión público que puede recibir tráfico de Internet, el tráfico entre clústeres entre los dominios sigue cifrado y seguro.

Eliminación de una conexión

La eliminación de una conexión detiene cualquier operación entre clústeres en sus índices.

1. En el panel del dominio, vaya a la pestaña Conexiones.
2. Seleccione las conexiones de dominio que desea eliminar, luego elija Eliminar y, a continuación, confirme la eliminación.

Puede realizar estos pasos en el dominio fuente o de destino para eliminar la conexión. Una vez quitada la conexión, seguirá visible con el estado `Deleted` durante un periodo de 15 días.

No se puede eliminar un dominio con conexiones activas entre clústeres. Para eliminar un dominio, primero quite todas las conexiones entrantes y salientes de ese dominio. Esto es para asegurarse de tener en cuenta a los usuarios del dominio entre clústeres antes de eliminar el dominio.

Configuración de seguridad y explicación de ejemplo

1. Envíe una petición de búsqueda en clústeres al dominio de origen.
2. El dominio de origen evalúa esa solicitud en comparación con la política de acceso al dominio. Dado que la búsqueda en clústeres requiere un control de acceso detallado, recomendamos una política de acceso abierto en el dominio de origen.

   JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
```

```
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "*"
        ]
      },
      "Action": [
        "es:ESHttp*"
      ],
      "Resource": "arn:aws:es:us-east-1:111122223333:domain/src-domain/*"
    }
  ]
}
```

> **Note**
>
> Si incluye índices remotos en la ruta, debe codificar la URL del URI en el ARN del dominio. Por ejemplo, use `arn:aws:es:us-east-1:123456789012:domain/my-domain/local_index,dst%3Aremote_index` en lugar de `arn:aws:es:us-east-1:123456789012:domain/my-domain/local_index,dst:remote_index`.

Si elige utilizar una política de acceso restrictivo además de un control de acceso detallado, la política debe permitir el acceso a `es:ESHttpGet` como mínimo.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "arn:aws:iam::111122223333:user/test-user"
        ]
      },
      "Action": "es:ESHttpGet",
      "Resource": "arn:aws:es:us-east-1:111122223333:domain/src-domain/*"
    }
```

```
  ]
}
```

3. El control de acceso detallado en el dominio de origen evalúa la solicitud:

   - ¿La solicitud está firmada con credenciales básicas IAM o HTTP válidas?
   - Si es así, ¿tiene el usuario permiso para realizar la búsqueda y acceder a los datos?

   Si la solicitud solo busca datos en el dominio de destino (por ejemplo, `dest-alias:dest-index/_search`), solo necesitará permisos en el dominio de destino.

   Si la solicitud busca datos en ambos dominios (por ejemplo, `source-index,dest-alias:dest-index/_search`), necesita permisos en ambos dominios.

   En el control de acceso detallado, los usuarios deben tener el permiso `indices:admin/shards/search_shards` además de los permisos `read` o `search` estándar para los índices pertinentes.

4. El dominio de origen pasa la solicitud al dominio de destino. El dominio de destino evalúa esta solicitud frente a su política de acceso al dominio. Debe incluir el permiso `es:ESCrossClusterGet` en el dominio de destino:

   JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
      },
      "Action": "es:ESCrossClusterGet",
      "Resource": "arn:aws:es:us-east-1:111122223333:domain/dst-domain"
    }
  ]
}
```

Asegúrese de que el permiso `es:ESCrossClusterGet` se aplica para `/dst-domain` y no `/dst-domain/*`.

Sin embargo, esta política mínima solo permite búsquedas en clústeres. Para realizar otras operaciones, como indexar documentos y realizar búsquedas estándar, necesita permisos adicionales. Recomendamos la siguiente política en el dominio de destino:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "*"
        ]
      },
      "Action": [
        "es:ESHttp*"
      ],
      "Resource": "arn:aws:es:us-east-1:111122223333:domain/dst-domain/*"
    },
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
      },
      "Action": "es:ESCrossClusterGet",
      "Resource": "arn:aws:es:us-east-1:111122223333:domain/dst-domain"
    }
  ]
}
```

> **Note**
>
> Todas las solicitudes de búsqueda en clústeres entre dominios se cifran en tránsito de forma predeterminada como parte del cifrado de nodo a nodo.

5. El dominio de destino realiza la búsqueda y devuelve los resultados al dominio de origen.
6. El dominio de origen combina sus propios resultados (si los hay) con los resultados del dominio de destino y los devuelve.
7. Recomendamos Postman para probar las solicitudes:

   - En el dominio de destino, indexe un documento:

   ```
   POST https://dst-domain.us-east-1.es.amazonaws.com/books/_doc/1

   {
     "Dracula": "Bram Stoker"
   }
   ```

   - Para consultar este índice desde el dominio de origen, incluya el alias de conexión del dominio de destino dentro de la consulta.

   ```
   GET https://src-domain.us-east-1.es.amazonaws.com/<connection_alias>:books/
   _search

   {
       ...
     "hits": [
       {
         "_index": "source-destination:books",
         "_type": "_doc",
         "_id": "1",
         "_score": 1,
         "_source": {
           "Dracula": "Bram Stoker"
         }
       }
     ]
   }
   ```

Puede encontrar el alias de conexión en la pestaña Conexiones del panel de control del dominio.

- Si configura una conexión entre `domain-a -> domain-b` con alias de conexión `cluster_b` y `domain-a -> domain-c` con alias de conexión `cluster_c`, busque `domain-a`, `domain-b` y `domain-c` de la siguiente manera:

```
GET https://src-domain.us-east-1.es.amazonaws.com/
local_index,cluster_b:b_index,cluster_c:c_index/_search
{
  "query": {
    "match": {
      "user": "domino"
    }
  }
}
```

Respuesta

```
{
  "took": 150,
  "timed_out": false,
  "_shards": {
    "total": 3,
    "successful": 3,
    "failed": 0,
    "skipped": 0
  },
  "_clusters": {
    "total": 3,
    "successful": 3,
    "skipped": 0
  },
  "hits": {
    "total": 3,
    "max_score": 1,
    "hits": [
      {
        "_index": "local_index",
        "_type": "_doc",
        "_id": "0",
        "_score": 1,
```

```
        "_source": {
          "user": "domino",
          "message": "This is message 1",
          "likes": 0
        }
      },
      {
        "_index": "cluster_b:b_index",
        "_type": "_doc",
        "_id": "0",
        "_score": 2,
        "_source": {
          "user": "domino",
          "message": "This is message 2",
          "likes": 0
        }
      },
      {
        "_index": "cluster_c:c_index",
        "_type": "_doc",
        "_id": "0",
        "_score": 3,
        "_source": {
          "user": "domino",
          "message": "This is message 3",
          "likes": 0
        }
      }
    ]
  }
}
```

Si no eligió omitir los clústeres no disponibles en la configuración de la conexión, todos los clústeres de destino que busca tienen que estar disponibles para que la petición de búsqueda se ejecute correctamente. De lo contrario, se produce un error en toda la solicitud; incluso si uno de los dominios no está disponible, la búsqueda no devuelve ningún resultado.

OpenSearch Dashboards

Puede visualizar datos de varios dominios conectados de la misma manera que desde un solo dominio, excepto que debe acceder a los índices remotos mediante `connection-alias:index`. Por lo tanto, su patrón de índice debe coincidir `connection-alias:index`.

Aprender a clasificar para Amazon OpenSearch Service

OpenSearch utiliza un marco de clasificación probabilístico llamado BM-25 para calcular las puntuaciones de relevancia. Si una palabra clave distintiva aparece con más frecuencia en un documento, BM-25 asigna una puntuación de relevancia más alta a ese documento. Este marco, sin embargo, no tiene en cuenta el comportamiento del usuario, como los datos de clics, lo que puede mejorar aún más la relevancia.

Learning to Rank es un complemento de código abierto que permite utilizar machine learning y los datos de comportamiento para ajustar la relevancia de los documentos. El complemento utiliza modelos de las bibliotecas XGBoost y Ranklib para volver a puntuar los resultados de búsqueda. El complemento de Elasticsearch LTR fue desarrollado inicialmente por OpenSource Connections, con importantes contribuciones de Wikimedia Foundation, Snagajob Engineering, Bonsai y Yelp Engineering. La versión de OpenSearch del plugin se deriva del complemento Elasticsearch LTR.

Aprender a clasificar requiere OpenSearch o Elasticsearch 7.7 o posterior. Para utilizar el complemento Aprender a clasificar, debe tener permisos de administrador completos. Para obtener más información, consulte the section called “Modificar el usuario maestro”.

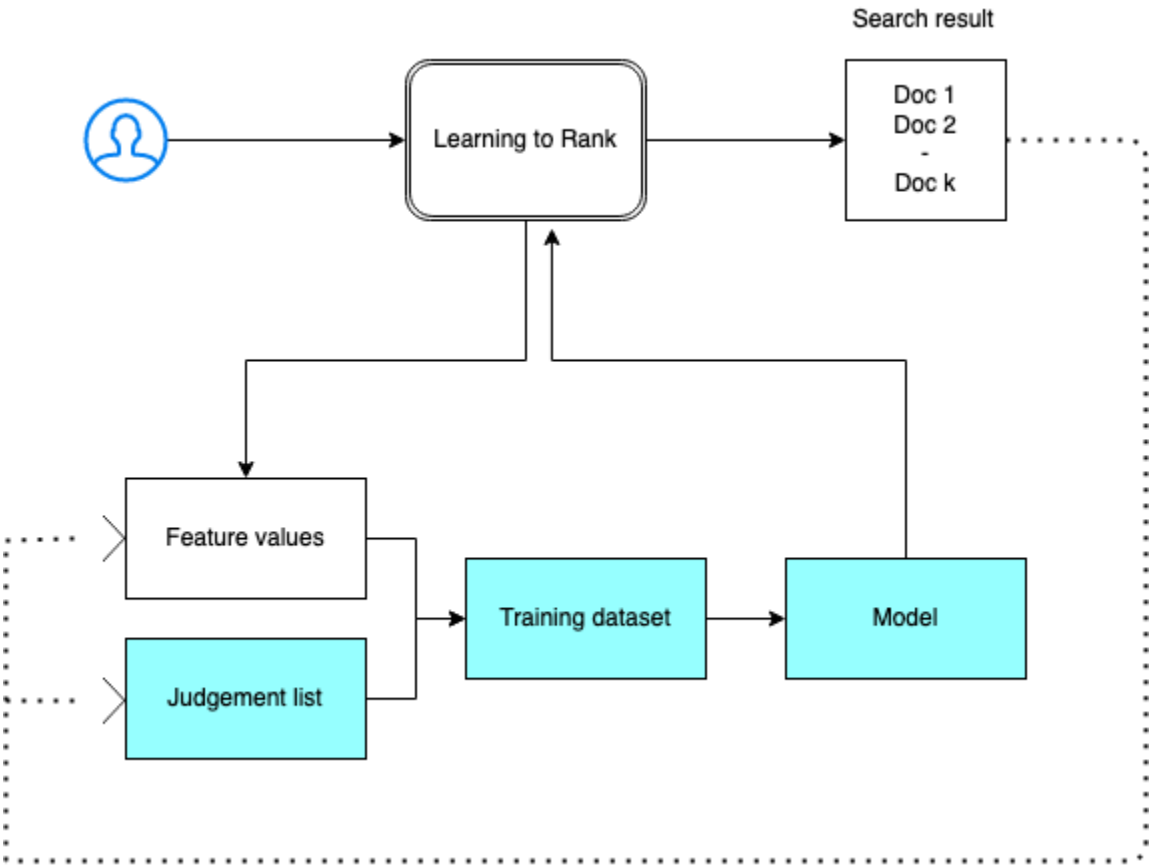
> **Note**
>
> Esta documentación proporciona información general del complemento Aprender a clasificar y le ayuda a empezar a usarlo. La documentación completa, incluidos los pasos detallados y las descripciones de la API, está disponible en la documentación de Learning to Rank (Aprender a clasificar).

Temas

- Introducción a Aprender a clasificar
- API de Aprender a clasificar

Introducción a Aprender a clasificar

Debe proporcionar una lista de criterios, preparar un conjunto de datos de entrenamiento y entrenar el modelo fuera de Amazon OpenSearch Service. Las partes en azul se producen fuera de OpenSearch Service:



Paso 1: inicializar el complemento

Para inicializar el complemento Aprender a clasificar, envíe la siguiente solicitud a su dominio de OpenSearch Service:

```
PUT _ltr
```

```
{
  "acknowledged" : true,
  "shards_acknowledged" : true,
  "index" : ".ltrstore"
}
```

Este comando crea un índice `.ltrstore` oculto que almacena información de metadatos, como conjuntos de características y modelos.

Paso 2: crear una lista de criterios

> **Note**
>
> Debe realizar este paso fuera de OpenSearch Service.

Una lista de criterios es un conjunto de ejemplos de los cuales aprende un modelo de machine learning. La lista de criterios debe incluir palabras clave importantes para usted y un conjunto de documentos calificados para cada palabra clave.

En este ejemplo, tenemos una lista de criterios para un conjunto de datos de películas. Un grado de 4 indica una coincidencia perfecta. Un grado de 0 indica la peor coincidencia.

Grado	Palabra clave	ID del documento	Nombre de la película
4	rambo	7555	Rambo
3	rambo	1370	Rambo III
3	rambo	1369	Rambo: First Blood Part II
3	rambo	1368	First Blood

Prepare la lista de criterios con el siguiente formato:

```
4 qid:1 # 7555 Rambo
3 qid:1 # 1370 Rambo III
3 qid:1 # 1369 Rambo: First Blood Part II
3 qid:1 # 1368 First Blood

where qid:1 represents "rambo"
```

Para ver un ejemplo más completo de una lista de criterios, consulte Criterios sobre películas.

Puede crear esta lista de criterios de forma manual con la ayuda de anotadores humanos o inferirla de manera programática a partir de datos de análisis.

Paso 3: crear un conjunto de características

Una característica es un campo que corresponde a la relevancia de un documento, por ejemplo, `title`, `overview`, `popularity score` (número de vistas) y así sucesivamente.

Cree un conjunto de características con una plantilla de Mustache para cada característica. Para obtener más información acerca de las características, consulte Trabajo con características.

En este ejemplo, crearemos un conjunto de características `movie_features` con los campos `title` y `overview`:

```
POST _ltr/_featureset/movie_features
{
  "featureset" : {
      "name" : "movie_features",
      "features" : [
        {
          "name" : "1",
          "params" : [
            "keywords"
          ],
          "template_language" : "mustache",
          "template" : {
            "match" : {
              "title" : "{{keywords}}"
            }
          }
        },
        {
          "name" : "2",
          "params" : [
            "keywords"
          ],
          "template_language" : "mustache",
          "template" : {
            "match" : {
              "overview" : "{{keywords}}"
            }
          }
        }
```

```
      ]
    }
}
```

Si consulta el índice `.ltrstore` original, recupera el conjunto de características:

```
GET _ltr/_featureset
```

Paso 4: registrar los valores de la característica

Los valores de la característica son las puntuaciones de relevancia calculadas por BM-25 para cada característica.

Combine el conjunto de características y la lista de sentencias para registrar los valores de la característica. Para obtener más información acerca de las características de registro, consulte Puntuaciones de características.

En este ejemplo, la consulta `bool` recupera los documentos calificados con el filtro y, a continuación, elige el conjunto de características con la consulta `sltr`. La consulta `ltr_log` combina los documentos y las características para registrar los valores de característica correspondientes:

```
POST tmdb/_search
{
  "_source": {
    "includes": [
      "title",
      "overview"
    ]
  },
  "query": {
    "bool": {
      "filter": [
        {
          "terms": {
            "_id": [
              "7555",
              "1370",
              "1369",
              "1368"
            ]
          }
        },
```

```
        {
          "sltr": {
            "_name": "logged_featureset",
            "featureset": "movie_features",
            "params": {
              "keywords": "rambo"
            }
          }
        }
      ]
    }
  },
  "ext": {
    "ltr_log": {
      "log_specs": {
        "name": "log_entry1",
        "named_query": "logged_featureset"
      }
    }
  }
}
```

Un ejemplo de respuesta se vería así:

```
{
  "took" : 7,
  "timed_out" : false,
  "_shards" : {
    "total" : 1,
    "successful" : 1,
    "skipped" : 0,
    "failed" : 0
  },
  "hits" : {
    "total" : {
      "value" : 4,
      "relation" : "eq"
    },
    "max_score" : 0.0,
    "hits" : [
      {
        "_index" : "tmdb",
        "_type" : "movie",
```

```
       "_id" : "1368",
       "_score" : 0.0,
       "_source" : {
         "overview" : "When former Green Beret John Rambo is harassed by local law
enforcement and arrested for vagrancy, the Vietnam vet snaps, runs for the hills and
rat-a-tat-tats his way into the action-movie hall of fame. Hounded by a relentless
sheriff, Rambo employs heavy-handed guerilla tactics to shake the cops off his tail.",
         "title" : "First Blood"
       },
       "fields" : {
         "_ltrlog" : [
           {
             "log_entry1" : [
               {
                 "name" : "1"
               },
               {
                 "name" : "2",
                 "value" : 10.558305
               }
             ]
           }
         ]
       },
       "matched_queries" : [
         "logged_featureset"
       ]
     },
     {
       "_index" : "tmdb",
       "_type" : "movie",
       "_id" : "7555",
       "_score" : 0.0,
       "_source" : {
         "overview" : "When governments fail to act on behalf of captive missionaries,
ex-Green Beret John James Rambo sets aside his peaceful existence along the Salween
River in a war-torn region of Thailand to take action.  Although he's still haunted
by violent memories of his time as a U.S. soldier during the Vietnam War, Rambo can
hardly turn his back on the aid workers who so desperately need his help.",
         "title" : "Rambo"
       },
       "fields" : {
         "_ltrlog" : [
           {
```

```
            "log_entry1" : [
              {
                "name" : "1",
                "value" : 11.2569065
              },
              {
                "name" : "2",
                "value" : 9.936821
              }
            ]
          }
        ]
      },
      "matched_queries" : [
        "logged_featureset"
      ]
    },
    {
      "_index" : "tmdb",
      "_type" : "movie",
      "_id" : "1369",
      "_score" : 0.0,
      "_source" : {
        "overview" : "Col. Troutman recruits ex-Green Beret John Rambo for a highly
secret and dangerous mission. Teamed with Co Bao, Rambo goes deep into Vietnam to
rescue POWs. Deserted by his own team, he's left in a hostile jungle to fight for his
life, avenge the death of a woman and bring corrupt officials to justice.",
        "title" : "Rambo: First Blood Part II"
      },
      "fields" : {
        "_ltrlog" : [
          {
            "log_entry1" : [
              {
                "name" : "1",
                "value" : 6.334839
              },
              {
                "name" : "2",
                "value" : 10.558305
              }
            ]
          }
        ]
```

```
        },
        "matched_queries" : [
          "logged_featureset"
        ]
      },
      {
        "_index" : "tmdb",
        "_type" : "movie",
        "_id" : "1370",
        "_score" : 0.0,
        "_source" : {
          "overview" : "Combat has taken its toll on Rambo, but he's finally begun to
 find inner peace in a monastery. When Rambo's friend and mentor Col. Trautman asks for
 his help on a top secret mission to Afghanistan, Rambo declines but must reconsider
 when Trautman is captured.",
          "title" : "Rambo III"
        },
        "fields" : {
          "_ltrlog" : [
            {
              "log_entry1" : [
                {
                  "name" : "1",
                  "value" : 9.425955
                },
                {
                  "name" : "2",
                  "value" : 11.262714
                }
              ]
            }
          ]
        },
        "matched_queries" : [
          "logged_featureset"
        ]
      }
    ]
  }
}
```

En el ejemplo anterior, la primera característica no posee un valor de característica, ya que la palabra clave “rambo” no aparece en el campo de título del documento con un ID igual a 1368. Se trata de un valor de característica que falta en los datos de entrenamiento.

Paso 5: crear un conjunto de datos de entrenamiento

> **Note**
>
> Debe realizar este paso fuera de OpenSearch Service.

El siguiente paso es combinar la lista de sentencias y los valores de característica para crear un conjunto de datos de entrenamiento. Si la lista de criterios original se ve así:

```
4 qid:1 # 7555 Rambo
3 qid:1 # 1370 Rambo III
3 qid:1 # 1369 Rambo: First Blood Part II
3 qid:1 # 1368 First Blood
```

Conviértala en el conjunto de datos de entrenamiento final, que se ve así:

```
4 qid:1 1:12.318474 2:10.573917 # 7555 rambo
3 qid:1 1:10.357875 2:11.950391 # 1370 rambo
3 qid:1 1:7.010513 2:11.220095 # 1369 rambo
3 qid:1 1:0.0 2:11.220095 # 1368 rambo
```

Puede realizar este paso de forma manual o escribir un programa para automatizarlo.

Paso 6: elegir un algoritmo y crear el modelo

> **Note**
>
> Debe realizar este paso fuera de OpenSearch Service.

Con el conjunto de datos de entrenamiento en su lugar, el siguiente paso es utilizar las bibliotecas XGBoost o Ranklib para crear un modelo. Las bibliotecas XGBoost y Ranklib permiten crear modelos populares como LambdaMART, Random Forests, etc.

Para conocer los pasos para utilizar XGBoost y Ranklib para crear el modelo, consulte la documentación sobre XGBoost y RankLib, respectivamente. Para utilizar Amazon SageMaker para crear el modelo XGBoost, consulte el Algoritmo XGBoost.

Paso 7: implementar el modelo

Después de haber creado el modelo, impleméntelo en el complemento Learning to Rank (Aprender a clasificar). Para obtener más información acerca de la implementación de un modelo, consulte Carga de un modelo entrenado.

En este ejemplo, crearemos un modelo de `my_ranklib_model` con la biblioteca Ranklib:

```
POST _ltr/_featureset/movie_features/_createmodel?pretty
{
  "model": {
    "name": "my_ranklib_model",
    "model": {
      "type": "model/ranklib",
      "definition": """## LambdaMART
## No. of trees = 10
## No. of leaves = 10
## No. of threshold candidates = 256
## Learning rate = 0.1
## Stop early = 100

<ensemble>
   <tree id="1" weight="0.1">
      <split>
         <feature>1</feature>
         <threshold>10.357875</threshold>
         <split pos="left">
            <feature>1</feature>
            <threshold>0.0</threshold>
            <split pos="left">
               <output>-2.0</output>
            </split>
            <split pos="right">
               <feature>1</feature>
               <threshold>7.010513</threshold>
               <split pos="left">
                  <output>-2.0</output>
               </split>
               <split pos="right">
```

```
                <output>-2.0</output>
             </split>
          </split>
       </split>
       <split pos="right">
          <output>2.0</output>
       </split>
    </split>
 </tree>
 <tree id="2" weight="0.1">
    <split>
       <feature>1</feature>
       <threshold>10.357875</threshold>
       <split pos="left">
          <feature>1</feature>
          <threshold>0.0</threshold>
          <split pos="left">
             <output>-1.67031991481781</output>
          </split>
          <split pos="right">
             <feature>1</feature>
             <threshold>7.010513</threshold>
             <split pos="left">
                <output>-1.67031991481781</output>
             </split>
             <split pos="right">
                <output>-1.6703200340270996</output>
             </split>
          </split>
       </split>
       <split pos="right">
          <output>1.6703201532363892</output>
       </split>
    </split>
 </tree>
 <tree id="3" weight="0.1">
    <split>
       <feature>2</feature>
       <threshold>10.573917</threshold>
       <split pos="left">
          <output>1.479954481124878</output>
       </split>
       <split pos="right">
          <feature>1</feature>
```

```
            <threshold>7.010513</threshold>
            <split pos="left">
               <feature>1</feature>
               <threshold>0.0</threshold>
               <split pos="left">
                  <output>-1.4799546003341675</output>
               </split>
               <split pos="right">
                  <output>-1.479954481124878</output>
               </split>
            </split>
            <split pos="right">
               <output>-1.479954481124878</output>
            </split>
         </split>
      </split>
   </tree>
   <tree id="4" weight="0.1">
      <split>
         <feature>1</feature>
         <threshold>10.357875</threshold>
         <split pos="left">
            <feature>1</feature>
            <threshold>0.0</threshold>
            <split pos="left">
               <output>-1.3569872379302979</output>
            </split>
            <split pos="right">
               <feature>1</feature>
               <threshold>7.010513</threshold>
               <split pos="left">
                  <output>-1.3569872379302979</output>
               </split>
               <split pos="right">
                  <output>-1.3569872379302979</output>
               </split>
            </split>
         </split>
         <split pos="right">
            <output>1.3569873571395874</output>
         </split>
      </split>
   </tree>
   <tree id="5" weight="0.1">
```

```
      <split>
         <feature>1</feature>
         <threshold>10.357875</threshold>
         <split pos="left">
            <feature>1</feature>
            <threshold>0.0</threshold>
            <split pos="left">
               <output>-1.2721362113952637</output>
            </split>
            <split pos="right">
               <feature>1</feature>
               <threshold>7.010513</threshold>
               <split pos="left">
                  <output>-1.2721363306045532</output>
               </split>
               <split pos="right">
                  <output>-1.2721363306045532</output>
               </split>
            </split>
         </split>
         <split pos="right">
            <output>1.2721362113952637</output>
         </split>
      </split>
   </tree>
   <tree id="6" weight="0.1">
      <split>
         <feature>1</feature>
         <threshold>10.357875</threshold>
         <split pos="left">
            <feature>1</feature>
            <threshold>7.010513</threshold>
            <split pos="left">
               <feature>1</feature>
               <threshold>0.0</threshold>
               <split pos="left">
                  <output>-1.2110036611557007</output>
               </split>
               <split pos="right">
                  <output>-1.2110036611557007</output>
               </split>
            </split>
            <split pos="right">
               <output>-1.2110037803649902</output>
```

```
            </split>
         </split>
         <split pos="right">
            <output>1.2110037803649902</output>
         </split>
      </split>
   </tree>
   <tree id="7" weight="0.1">
      <split>
         <feature>1</feature>
         <threshold>10.357875</threshold>
         <split pos="left">
            <feature>1</feature>
            <threshold>7.010513</threshold>
            <split pos="left">
               <feature>1</feature>
               <threshold>0.0</threshold>
               <split pos="left">
                  <output>-1.165616512298584</output>
               </split>
               <split pos="right">
                  <output>-1.165616512298584</output>
               </split>
            </split>
            <split pos="right">
               <output>-1.165616512298584</output>
            </split>
         </split>
         <split pos="right">
            <output>1.165616512298584</output>
         </split>
      </split>
   </tree>
   <tree id="8" weight="0.1">
      <split>
         <feature>1</feature>
         <threshold>10.357875</threshold>
         <split pos="left">
            <feature>1</feature>
            <threshold>7.010513</threshold>
            <split pos="left">
               <feature>1</feature>
               <threshold>0.0</threshold>
               <split pos="left">
```

```
                <output>-1.131177544593811</output>
             </split>
             <split pos="right">
                <output>-1.131177544593811</output>
             </split>
          </split>
          <split pos="right">
             <output>-1.131177544593811</output>
          </split>
       </split>
       <split pos="right">
          <output>1.131177544593811</output>
       </split>
    </split>
 </tree>
 <tree id="9" weight="0.1">
    <split>
       <feature>2</feature>
       <threshold>10.573917</threshold>
       <split pos="left">
          <output>1.1046180725097656</output>
       </split>
       <split pos="right">
          <feature>1</feature>
          <threshold>7.010513</threshold>
          <split pos="left">
             <feature>1</feature>
             <threshold>0.0</threshold>
             <split pos="left">
                <output>-1.1046180725097656</output>
             </split>
             <split pos="right">
                <output>-1.1046180725097656</output>
             </split>
          </split>
          <split pos="right">
             <output>-1.1046180725097656</output>
          </split>
       </split>
    </split>
 </tree>
 <tree id="10" weight="0.1">
    <split>
       <feature>1</feature>
```

```
         <threshold>10.357875</threshold>
         <split pos="left">
            <feature>1</feature>
            <threshold>7.010513</threshold>
            <split pos="left">
               <feature>1</feature>
               <threshold>0.0</threshold>
               <split pos="left">
                  <output>-1.0838804244995117</output>
               </split>
               <split pos="right">
                  <output>-1.0838804244995117</output>
               </split>
            </split>
            <split pos="right">
               <output>-1.0838804244995117</output>
            </split>
         </split>
         <split pos="right">
            <output>1.0838804244995117</output>
         </split>
      </split>
   </tree>
</ensemble>
"""
    }
  }
}
```

Para ver el modelo, envíe la siguiente solicitud:

```
GET _ltr/_model/my_ranklib_model
```

Paso 8: buscar con Aprender a clasificar

Después de implementar el modelo, estará listo para realizar la búsqueda.

Realice la consulta `sltr` con las características que utiliza y el nombre del modelo que desea ejecutar:

```
POST tmdb/_search
{
```

```
  "_source": {
    "includes": ["title", "overview"]
  },
  "query": {
    "multi_match": {
      "query": "rambo",
      "fields": ["title", "overview"]
    }
  },
  "rescore": {
    "query": {
      "rescore_query": {
        "sltr": {
          "params": {
            "keywords": "rambo"
          },
          "model": "my_ranklib_model"
        }
      }
    }
  }
}
```

Con Aprender a clasificar, usted ve “Rambo” como el primer resultado porque le asignamos la calificación más alta en la lista de criterios:

```
{
  "took" : 12,
  "timed_out" : false,
  "_shards" : {
    "total" : 1,
    "successful" : 1,
    "skipped" : 0,
    "failed" : 0
  },
  "hits" : {
    "total" : {
      "value" : 7,
      "relation" : "eq"
    },
    "max_score" : 13.096414,
    "hits" : [
      {
```

```
        "_index" : "tmdb",
        "_type" : "movie",
        "_id" : "7555",
        "_score" : 13.096414,
        "_source" : {
          "overview" : "When governments fail to act on behalf of captive missionaries,
ex-Green Beret John James Rambo sets aside his peaceful existence along the Salween
River in a war-torn region of Thailand to take action.  Although he's still haunted
by violent memories of his time as a U.S. soldier during the Vietnam War, Rambo can
hardly turn his back on the aid workers who so desperately need his help.",
          "title" : "Rambo"
        }
      },
      {
        "_index" : "tmdb",
        "_type" : "movie",
        "_id" : "1370",
        "_score" : 11.17245,
        "_source" : {
          "overview" : "Combat has taken its toll on Rambo, but he's finally begun to
find inner peace in a monastery. When Rambo's friend and mentor Col. Trautman asks for
his help on a top secret mission to Afghanistan, Rambo declines but must reconsider
when Trautman is captured.",
          "title" : "Rambo III"
        }
      },
      {
        "_index" : "tmdb",
        "_type" : "movie",
        "_id" : "1368",
        "_score" : 10.442155,
        "_source" : {
          "overview" : "When former Green Beret John Rambo is harassed by local law
enforcement and arrested for vagrancy, the Vietnam vet snaps, runs for the hills and
rat-a-tat-tats his way into the action-movie hall of fame. Hounded by a relentless
sheriff, Rambo employs heavy-handed guerilla tactics to shake the cops off his tail.",
          "title" : "First Blood"
        }
      },
      {
        "_index" : "tmdb",
        "_type" : "movie",
        "_id" : "1369",
        "_score" : 10.442155,
```

```
      "_source" : {
        "overview" : "Col. Troutman recruits ex-Green Beret John Rambo for a highly
secret and dangerous mission. Teamed with Co Bao, Rambo goes deep into Vietnam to
rescue POWs. Deserted by his own team, he's left in a hostile jungle to fight for his
life, avenge the death of a woman and bring corrupt officials to justice.",
        "title" : "Rambo: First Blood Part II"
      }
    },
    {
      "_index" : "tmdb",
      "_type" : "movie",
      "_id" : "31362",
      "_score" : 7.424202,
      "_source" : {
        "overview" : "It is 1985, and a small, tranquil Florida town is being rocked
by a wave of vicious serial murders and bank robberies. Particularly sickening to the
authorities is the gratuitous use of violence by two “Rambo” like killers who dress
themselves in military garb. Based on actual events taken from FBI files, the movie
depicts the Bureau’s efforts to track down these renegades.",
        "title" : "In the Line of Duty: The F.B.I. Murders"
      }
    },
    {
      "_index" : "tmdb",
      "_type" : "movie",
      "_id" : "13258",
      "_score" : 6.43182,
      "_source" : {
        "overview" : """Will Proudfoot (Bill Milner) is looking for an escape from
his family's stifling home life when he encounters Lee Carter (Will Poulter), the
school bully. Armed with a video camera and a copy of "Rambo: First Blood", Lee plans
to make cinematic history by filming his own action-packed video epic. Together, these
two newfound friends-turned-budding-filmmakers quickly discover that their imaginative
– and sometimes mishap-filled – cinematic adventure has begun to take on a life of its
own!""",
        "title" : "Son of Rambow"
      }
    },
    {
      "_index" : "tmdb",
      "_type" : "movie",
      "_id" : "61410",
      "_score" : 3.9719706,
      "_source" : {
```

```
          "overview" : "It's South Africa 1990. Two major events are about to happen:
 The release of Nelson Mandela and, more importantly, it's Spud Milton's first year
 at an elite boys only private boarding school. John Milton is a boy from an ordinary
 background who wins a scholarship to a private school in Kwazulu-Natal, South Africa.
 Surrounded by boys with nicknames like Gecko, Rambo, Rain Man and Mad Dog, Spud has
 his hands full trying to adapt to his new home. Along the way Spud takes his first
 tentative steps along the path to manhood. (The path it seems could be a rather long
 road). Spud is an only child. He is cursed with parents from well beyond the lunatic
 fringe and a senile granny. His dad is a fervent anti-communist who is paranoid that
 the family domestic worker is running a shebeen from her room at the back of the
 family home. His mom is a free spirit and a teenager's worst nightmare, whether it's
 shopping for Spud's underwear in the local supermarket",
          "title" : "Spud"
        }
      }
    ]
  }
}
```

Si realiza la búsqueda sin utilizar el complemento Aprender a clasificar, OpenSearch devuelve resultados diferentes:

```
POST tmdb/_search
{
  "_source": {
    "includes": ["title", "overview"]
  },
  "query": {
    "multi_match": {
      "query": "Rambo",
      "fields": ["title", "overview"]
    }
  }
}
```

```
{
  "took" : 5,
  "timed_out" : false,
  "_shards" : {
    "total" : 1,
    "successful" : 1,
    "skipped" : 0,
    "failed" : 0
```

```
 },
 "hits" : {
   "total" : {
     "value" : 5,
     "relation" : "eq"
   },
   "max_score" : 11.262714,
   "hits" : [
     {
       "_index" : "tmdb",
       "_type" : "movie",
       "_id" : "1370",
       "_score" : 11.262714,
       "_source" : {
         "overview" : "Combat has taken its toll on Rambo, but he's finally begun to
find inner peace in a monastery. When Rambo's friend and mentor Col. Trautman asks for
his help on a top secret mission to Afghanistan, Rambo declines but must reconsider
when Trautman is captured.",
         "title" : "Rambo III"
       }
     },
     {
       "_index" : "tmdb",
       "_type" : "movie",
       "_id" : "7555",
       "_score" : 11.2569065,
       "_source" : {
         "overview" : "When governments fail to act on behalf of captive missionaries,
ex-Green Beret John James Rambo sets aside his peaceful existence along the Salween
River in a war-torn region of Thailand to take action.  Although he's still haunted
by violent memories of his time as a U.S. soldier during the Vietnam War, Rambo can
hardly turn his back on the aid workers who so desperately need his help.",
         "title" : "Rambo"
       }
     },
     {
       "_index" : "tmdb",
       "_type" : "movie",
       "_id" : "1368",
       "_score" : 10.558305,
       "_source" : {
         "overview" : "When former Green Beret John Rambo is harassed by local law
enforcement and arrested for vagrancy, the Vietnam vet snaps, runs for the hills and
```

```
 rat-a-tat-tats his way into the action-movie hall of fame. Hounded by a relentless
 sheriff, Rambo employs heavy-handed guerilla tactics to shake the cops off his tail.",
          "title" : "First Blood"
        }
      },
      {
        "_index" : "tmdb",
        "_type" : "movie",
        "_id" : "1369",
        "_score" : 10.558305,
        "_source" : {
          "overview" : "Col. Troutman recruits ex-Green Beret John Rambo for a highly
 secret and dangerous mission. Teamed with Co Bao, Rambo goes deep into Vietnam to
 rescue POWs. Deserted by his own team, he's left in a hostile jungle to fight for his
 life, avenge the death of a woman and bring corrupt officials to justice.",
          "title" : "Rambo: First Blood Part II"
        }
      },
      {
        "_index" : "tmdb",
        "_type" : "movie",
        "_id" : "13258",
        "_score" : 6.4600153,
        "_source" : {
          "overview" : """Will Proudfoot (Bill Milner) is looking for an escape from
 his family's stifling home life when he encounters Lee Carter (Will Poulter), the
 school bully. Armed with a video camera and a copy of "Rambo: First Blood", Lee plans
 to make cinematic history by filming his own action-packed video epic. Together, these
 two newfound friends-turned-budding-filmmakers quickly discover that their imaginative
 – and sometimes mishap-filled – cinematic adventure has begun to take on a life of its
 own!""",
          "title" : "Son of Rambow"
        }
      }
    ]
  }
}
```

En función de cuán bien considere que el modelo funciona, ajuste la lista de criterios y las características. A continuación, repita los pasos 2 a 8 para mejorar los resultados de la clasificación a lo largo del tiempo.

API de Aprender a clasificar

Utilice las operaciones de Aprender a clasificar para trabajar de manera programática con conjuntos de características y modelos.

Crear tienda

Crea un índice `.ltrstore` oculto que almacena información sobre metadatos, como conjuntos de características y modelos.

```
PUT _ltr
```

Eliminar tienda

Elimina el índice `.ltrstore` oculto y restablece el complemento.

```
DELETE _ltr
```

Crear conjunto de características

Crea un conjunto de características.

```
POST _ltr/_featureset/<name_of_features>
```

Eliminar un conjunto de características

Elimina un conjunto de características.

```
DELETE _ltr/_featureset/<name_of_feature_set>
```

Obtener un conjunto de características

Recupera un conjunto de características.

```
GET _ltr/_featureset/<name_of_feature_set>
```

Crear un modelo

Crea un modelo.

```
POST _ltr/_featureset/<name_of_feature_set>/_createmodel
```

Eliminar modelo

Elimina un modelo.

```
DELETE _ltr/_model/<name_of_model>
```

Obtener modelo

Recupera un modelo.

```
GET _ltr/_model/<name_of_model>
```

Obtener estadísticas

Proporciona información acerca de cómo se comporta el complemento.

```
GET _ltr/_stats
```

También puede usar filtros para recuperar una sola estadística:

```
GET _ltr/_stats/<stat>
```

Además, puede limitar la información a un único nodo en el clúster:

```
GET _ltr/_stats/<stat>/nodes/<nodeId>

{
  "_nodes" : {
    "total" : 1,
    "successful" : 1,
    "failed" : 0
  },
  "cluster_name" : "873043598401:ltr-77",
  "stores" : {
    ".ltrstore" : {
      "model_count" : 1,
      "featureset_count" : 1,
      "feature_count" : 2,
```

```
      "status" : "green"
    }
  },
  "status" : "green",
  "nodes" : {
    "DjelK-_ZSfyzstO5dhGGQA" : {
      "cache" : {
        "feature" : {
          "eviction_count" : 0,
          "miss_count" : 0,
          "entry_count" : 0,
          "memory_usage_in_bytes" : 0,
          "hit_count" : 0
        },
        "featureset" : {
          "eviction_count" : 2,
          "miss_count" : 2,
          "entry_count" : 0,
          "memory_usage_in_bytes" : 0,
          "hit_count" : 0
        },
        "model" : {
          "eviction_count" : 2,
          "miss_count" : 3,
          "entry_count" : 1,
          "memory_usage_in_bytes" : 3204,
          "hit_count" : 1
        }
      },
      "request_total_count" : 6,
      "request_error_count" : 0
    }
  }
}
```

Las estadísticas se proporcionan en dos niveles, nodo y clúster, como se especifica en las siguientes tablas:

Estadísticas a nivel de nodo

Nombre del campo	Descripción
request_total_count	Recuento total de solicitudes de clasificación.

Nombre del campo	Descripción
request_error_count	Recuento total de solicitudes fallidas.
memoria caché	Estadísticas de todas las cachés (caracter ísticas, conjuntos de características, modelos). Se produce un acierto de caché cuando un usuario consulta el complemento y el modelo ya está cargado en la memoria.
cache.eviction_count	Número de expulsiones de caché.
cache.hit_count	Número de aciertos de caché.
cache.miss_count	Número de errores de caché. Se produce un error de caché cuando un usuario consulta el complemento y el modelo aún no se cargó en la memoria.
cache.entry_count	Número de entradas en la caché.
cache.memory_usage_in_bytes	Memoria total utilizada en bytes.
cache.cache_capacity_reached	Indica si se alcanza el límite de caché.

Estadísticas de nivel de clúster

Nombre del campo	Descripción
tiendas	Indica dónde se almacenan los conjuntos de características y los metadatos del modelo. (El valor predeterminado es “.ltrstore”. De lo contrario, tiene el prefijo “.ltrstore_”, con un nombre proporcionado por el usuario).
stores.status	Estado del índice.
stores.feature_sets	Número de conjuntos de características.

Nombre del campo	Descripción
stores.features_count	Número de características.
stores.model_count	Número de modelos.
status	Estado del complemento basado en el estado de los índices del almacén de características (rojo, amarillo o verde) y el estado del interrupt or de circuito (abierto o cerrado).
cache.cache_capacity_reached	Indica si se alcanza el límite de caché.

Obtener estadísticas de caché

Devuelve estadísticas sobre el uso de la memoria y la caché.

```
GET _ltr/_cachestats

{
    "_nodes": {
        "total": 2,
        "successful": 2,
        "failed": 0
    },
    "cluster_name": "opensearch-cluster",
    "all": {
        "total": {
            "ram": 612,
            "count": 1
        },
        "features": {
            "ram": 0,
            "count": 0
        },
        "featuresets": {
            "ram": 612,
            "count": 1
        },
        "models": {
            "ram": 0,
```

```
            "count": 0
        }
    },
    "stores": {
        ".ltrstore": {
            "total": {
                "ram": 612,
                "count": 1
            },
            "features": {
                "ram": 0,
                "count": 0
            },
            "featuresets": {
                "ram": 612,
                "count": 1
            },
            "models": {
                "ram": 0,
                "count": 0
            }
        }
    },
    "nodes": {
        "ejF6uutERF20wOFNOXB61A": {
            "name": "opensearch1",
            "hostname": "172.18.0.4",
            "stats": {
                "total": {
                    "ram": 612,
                    "count": 1
                },
                "features": {
                    "ram": 0,
                    "count": 0
                },
                "featuresets": {
                    "ram": 612,
                    "count": 1
                },
                "models": {
                    "ram": 0,
                    "count": 0
                }
```

```
            }
        },
        "Z2RZNWRLSveVcz2c6lHf5A": {
            "name": "opensearch2",
            "hostname": "172.18.0.2",
            "stats": {
                ...
            }
        }
    }
}
```

Borrar caché

Borra la caché del complemento. Utilice esta opción para actualizar el modelo.

```
POST _ltr/_clearcache
```

Búsqueda asíncrona en Amazon OpenSearch Service

Con la búsqueda asíncrona de Amazon OpenSearch Service, puede enviar una consulta de búsqueda que se ejecute en segundo plano, monitorear el progreso de la solicitud y recuperar los resultados en una etapa posterior. Puede recuperar resultados parciales, a medida que estén disponibles, antes de que se complete la búsqueda. Una vez finalizada la búsqueda, guarde los resultados para una posterior recuperación y análisis.

La búsqueda asíncrona requiere OpenSearch 1.0 o posterior o Elasticsearch 7.10 o posterior.

Esta documentación proporciona una breve información general de la búsqueda asíncrona. También se analizan las limitaciones del uso de la búsqueda asíncrona con un dominio administrado de Amazon OpenSearch Service en lugar de con un clúster de OpenSearch de código abierto. Para obtener la documentación completa de la búsqueda asíncrona, que incluye la configuración disponible, los permisos y una referencia completa de la API, consulte Asynchronous search en la documentación de OpenSearch.

Ejemplo de llamada de búsqueda

Para realizar una búsqueda asíncrona, envíe solicitudes HTTP a `_plugins/_asynchronous_search` con el siguiente formato:

```
POST opensearch-domain/_plugins/_asynchronous_search
```

> **Note**
>
> Si utiliza Elasticsearch 7.10 en lugar de una versión de OpenSearch, reemplace `_plugins` con `_opendistro` en todas las solicitudes de búsqueda asíncronas.

Puede especificar las siguientes opciones de búsqueda asíncrona:

Opciones	Descripción	Valor predeterminado	Obligatorio
`wait_for_completion_timeout`	Especifica la cantidad de tiempo durante la cual planea esperar los resultados. Puede ver los resultados que obtiene dentro de este tiempo, igual que en una búsqueda normal. Puede sondear los resultados restantes en función de un ID. El valor máximo es 300 segundos.	1 segundos	No
`keep_on_completion`	Especifica si desea guardar los resultados en el clúster una vez finalizada la búsqueda. Puede examinar los resultados almacenados más adelante.	`false`	No
`keep_alive`	Especifica la cantidad de tiempo que se guarda el resultado en el clúster. Por ejemplo, `2d` significa que los resultados se almacenan en el clúster durante 48 horas. Los resultados de búsqueda guardados se eliminan después de este periodo o si se cancela la búsqueda. Tenga en cuenta que esto incluye el tiempo de ejecución de la consulta. Si la consulta sobrepasa este tiempo, el proceso cancela esta consulta automáticamente.	12 horas	No

Solicitud de ejemplo

```
POST _plugins/_asynchronous_search/?
pretty&size=10&wait_for_completion_timeout=1ms&keep_on_completion=true&request_cache=false
{
  "aggs": {
    "city": {
      "terms": {
        "field": "city",
        "size": 10
      }
    }
  }
}
```

> **Note**
>
> Todos los parámetros de solicitud que se aplican a una consulta `_search` estándar son compatibles. Si utiliza Elasticsearch 7.10 en lugar de una versión de OpenSearch, reemplace `_plugins` con `_opendistro`.

Permisos de búsqueda asíncrona

La búsqueda asíncrona admite el control de acceso detallado. Para obtener más información sobre la combinación y la coincidencia de permisos para adaptarlos al caso de uso, consulte Asynchronous search security.

Para los dominios con control de acceso detallado habilitado, necesita los siguientes permisos mínimos para un rol:

```
# Allows users to use all asynchronous search functionality
asynchronous_search_full_access:
  reserved: true
  cluster_permissions:
    - 'cluster:admin/opensearch/asynchronous-search/*'
  index_permissions:
    - index_patterns:
        - '*'
      allowed_actions:
        - 'indices:data/read/search*'
```

```
# Allows users to read stored asynchronous search results
asynchronous_search_read_access:
  reserved: true
  cluster_permissions:
    - 'cluster:admin/opensearch/asynchronous-search/get'
```

Para dominios con control de acceso detallado deshabilitado, utilice el acceso de IAM y la clave secreta para firmar todas las solicitudes. Puede acceder a los resultados con el ID de búsqueda asíncrona.

Configuración de búsqueda asincrónica

OpenSearch permite cambiar todas las configuraciones de búsqueda asincrónica mediante la API de `_cluster/settings`. En OpenSearch Service, solo puede cambiar la configuración siguiente:

- `plugins.asynchronous_search.node_concurrent_running_searches`
- `plugins.asynchronous_search.persist_search_failures`

Búsqueda en clústeres

Puede realizar una búsqueda asíncrona entre clústeres con las siguientes limitaciones menores:

- Puede ejecutar una búsqueda asíncrona solo en el dominio fuente.
- No puede minimizar los viajes de ida y vuelta de la red como parte de una consulta de búsqueda entre clústeres.

Si configura una conexión entre `domain-a -> domain-b`, con alias de conexión `cluster_b`, y `domain-a -> domain-c`, con alias de conexión `cluster_c`, realice una búsqueda asíncrona de `domain-a`, `domain-b` y `domain-c`, de la siguiente manera:

```
POST https://src-domain.us-east-1.es.amazonaws.com/
local_index,cluster_b:b_index,cluster_c:c_index/_plugins/_asynchronous_search/?
pretty&size=10&wait_for_completion_timeout=500ms&keep_on_completion=true&request_cache=false
{
  "size": 0,
  "_source": {
    "excludes": []
  },
```

```
  "aggs": {
    "2": {
      "terms": {
        "field": "clientip",
        "size": 50,
        "order": {
          "_count": "desc"
        }
      }
    }
  },
  "stored_fields": [
    "*"
  ],
  "script_fields": {},
  "docvalue_fields": [
    "@timestamp"
  ],
  "query": {
    "bool": {
      "must": [
        {
          "query_string": {
            "query": "status:404",
            "analyze_wildcard": true,
            "default_field": "*"
          }
        },
        {
          "range": {
            "@timestamp": {
              "gte": 1483747200000,
              "lte": 1488326400000,
              "format": "epoch_millis"
            }
          }
        }
      ],
      "filter": [],
      "should": [],
      "must_not": []
    }
  }
```

```
}
```

Respuesta

```
{
  "id" :
 "Fm9pYzJyVG91U19xb0hIQUJnMHJfRFEAAAAAAAknghQ1OWVBczNZQjVEa2dMYTBXaTdEagAAAAAAAAAB",
  "state" : "RUNNING",
  "start_time_in_millis" : 1609329314796,
  "expiration_time_in_millis" : 1609761314796
}
```

Para obtener más información, consulte the section called “Búsqueda en clústeres”.

UltraWarm

Las búsquedas asíncronas con índices de UltraWarm siguen en funcionamiento. Para obtener más información, consulte the section called “Almacenamiento UltraWarm”.

> **Note**
>
> Puede monitorear las estadísticas de búsqueda asíncrona en CloudWatch. Para obtener una lista completa de las métricas, consulte the section called “Métricas de búsqueda asíncrona”.

Búsqueda de punto en el tiempo en Amazon OpenSearch Service

El punto en el tiempo (PIT) es un tipo de búsqueda que permite ejecutar diferentes consultas en un conjunto de datos fijo en el tiempo. Normalmente, cuando ejecuta la misma consulta en el mismo índice en distintos puntos en el tiempo, recibe resultados diferentes porque los documentos se indexan, actualizan y eliminan constantemente. Con PIT, puede realizar consultas en función de un estado constante de su conjunto de datos.

El uso principal de la búsqueda es combinarla con la funcionalidad `search_after`. Este es el método de paginación preferido en OpenSearch, especialmente para la paginación profunda, porque funciona en un conjunto de datos que está congelado en el tiempo, no está vinculado a una consulta y admite una paginación coherente hacia adelante y hacia atrás. Puede usar PIT con un dominio que ejecute la versión 2.5 de OpenSearch.

> **Note**
>
> En este tema se proporciona información general sobre PIT y algunos aspectos que se deben tener en cuenta al utilizarlo en un dominio administrado de Amazon OpenSearch Service en lugar de en un clúster de OpenSearch autoadministrado. Para obtener la documentación completa de PIT, que incluye una referencia exhaustiva sobre la API, consulte Point in Time en la documentación de código abierto de OpenSearch.

Consideraciones

Al configurar las búsquedas PIT, tenga en cuenta lo siguiente:

- Si va a llevar a cabo una actualización desde un dominio que ejecuta la versión 2.3 de OpenSearch y necesita un control de acceso detallado para las acciones PIT, tendrá que agregar esas acciones y funciones manualmente.
- El PIT no tiene capacidad de recuperación. El reinicio de los nodos, la terminación de nodos, las implementaciones azul/verde y los reinicios de los procesos de OpenSearch provocan la pérdida de todos los datos de PIT.
- Si una partición se reubica durante la implementación azul/verde, solo los segmentos de datos activos se transfieren al nuevo nodo. Los segmentos de particiones retenidos por PIT (tanto de forma exclusiva como el que se comparte con los datos activos) permanecen en el nodo anterior.
- Actualmente, las búsquedas PIT no funcionan con la búsqueda asíncrona.

Creación de un PIT

Para ejecutar una consulta de PIT, envíe solicitudes HTTP a `_search/point_in_time` con el siguiente formato:

```
POST opensearch-domain/my-index/_search/point_in_time?keep_alive=time
```

Puede especificar las siguientes opciones de PIT:

Opciones	Descripción	Valor predeterminado	Obligatorio
`keep_alive`	La cantidad de tiempo que debe durar la conservación del PIT. Cada vez que se accede a un PIT con una solicitud de búsqueda, la vida útil del PIT se prolonga en un tiempo igual al parámetro `keep_alive` . Este parámetro de consulta es obligatorio cuando se crea un PIT, pero es opcional en una solicitud de búsqueda.		Sí
`preference`	Una cadena que especifica el nodo o la partición utilizados para realizar la búsqueda.	Random	No
`routing`	Una cadena que especifica que las solicitudes de búsqueda se dirijan a una partición específica.	El `_id` del documento	No
`expand_wildcards`	Una cadena que especifica el tipo de índice que puede coincidir con el patrón comodín. Admite valores separados por comas. Los valores válidos son los siguientes: • `all`: Coincide con cualquier índice o flujo de datos, incluidos los ocultos. • `open`: Coincide con índices abiertos, no ocultos o flujos de datos no ocultos. • `closed`: Coincide con índices cerrados, no ocultos o flujos de datos no ocultos. • `hidden`: Coincide con índices o flujos de datos ocultos. Debe combinarse con índices abiertos, cerrados o tanto abiertos como cerrados. • `none`: No se aceptan patrones comodín.	`open`	No

Opciones	Descripción	Valor predeterminado	Obligatorio
`allow_partial_pit_creation`	Un valor booleano que especifica si se debe crear un PIT con errores parciales.	`true`	No

Respuesta de ejemplo

```
{
    "pit_id":
 "o463QQEPbXktaW5kZXgtMDAwMDAxFnNOWU43ckt3U3IyaFVpbGE1UWEtMncAFjFyeXBsRGJmVFM2RTB6eVg1aVVqQncAA
    "_shards": {
        "total": 1,
        "successful": 1,
        "skipped": 0,
        "failed": 0
    },
    "creation_time": 1658146050064
}
```

Al crear un PIT, recibirá un ID de PIT en la respuesta. Este es el ID que se utiliza para realizar búsquedas con el PIT.

Permisos de puntos en el tiempo

Los PIT admiten el control de acceso detallado. Si va a actualizar a un dominio con la versión 2.5 de OpenSearch y necesita un control de acceso detallado, debe crear roles manualmente con los siguientes permisos:

```
# Allows users to use all point in time search search functionality
point_in_time_full_access:
  reserved: true
  index_permissions:
    - index_patterns:
        - '*'
      allowed_actions:
        - "indices:data/read/point_in_time/create"
        - "indices:data/read/point_in_time/delete"
```

```
        - "indices:data/read/point_in_time/readall"
        - "indices:data/read/search"
        - "indices:monitor/point_in_time/segments"


# Allows users to use point in time search search functionality for specific index
# All type operations like list all PITs, delete all PITs are not supported in this
 case

point_in_time_index_access:
  reserved: true
  index_permissions:
    - index_patterns:
        - 'my-index-1'
      allowed_actions:
        - "indices:data/read/point_in_time/create"
        - "indices:data/read/point_in_time/delete"
        - "indices:data/read/search"
        - "indices:monitor/point_in_time/segments"
```

Para los dominios con la versión 2.5 o posterior de OpenSearch, puede usar el rol integrado `point_in_time_full_access`. Para obtener más información, consulte el Modelo de seguridad en la documentación de OpenSearch.

Configuración de PIT

OpenSearch permite cambiar todas las configuraciones de PIT disponibles mediante la API de `_cluster/settings`. En OpenSearch Service, actualmente no es posible modificar la configuración.

Búsqueda en clústeres

Puede crear PIT, buscarlas con identificadores de PIT, enumerarlas y eliminarlas de todos los clústeres con las siguientes limitaciones menores:

- Puede enumerar todos los PIT y eliminarlos solo en el dominio origen.
- No puede minimizar los viajes de ida y vuelta de la red como parte de una consulta de búsqueda entre clústeres.

Para obtener más información, consulte the section called “Búsqueda en clústeres”.

UltraWarm

Las búsquedas PIT con índices de UltraWarm siguen en funcionamiento. Para obtener más información, consulte the section called “Almacenamiento UltraWarm”.

> **Note**
>
> Puede monitorear las estadísticas de búsqueda PIT en CloudWatch. Para obtener una lista completa de las métricas, consulte the section called “Métricas de un momento dado”.

Búsqueda semántica en Amazon OpenSearch Service

A partir de la versión 2.9 de OpenSearch, puede utilizar la búsqueda semántica para comprender las consultas de búsqueda y mejorar la relevancia de las búsquedas. Puede usar la búsqueda semántica de dos maneras: con la búsqueda neuronal y con la búsqueda de k-Nearest Neighbor (k-NN).

Con OpenSearch Service, puede configurar conectores de IA para Servicios de AWS y servicios externos. Con la consola, también puede crear un modelo de ML con una plantilla CloudFormation. Para obtener más información, consulte the section called “Integraciones de plantilla de CloudFormation”.

Para obtener la documentación completa sobre la búsqueda semántica, que incluye una guía paso a paso para usar la búsqueda semántica, consulte Semantic search en la documentación de código abierto de OpenSearch.

Búsqueda de segmentos simultánea en Amazon OpenSearch Service

A partir de la versión 2.17 de OpenSearch, la búsqueda simultánea de segmentos utiliza un nuevo ajuste para controlar el comportamiento de la búsqueda concurrente.

- Los dominios nuevos creados con la versión 2.17 tienen la búsqueda simultánea de segmentos configurada en modo automático de manera predeterminada en los nodos de tamaño 2xl o superior.
- Los dominios existentes que se actualizan a 2.17 tienen la búsqueda simultánea de segmentos configurada en modo automático según el tipo de instancia para todos los nodos de tamaño 2xl o superior y si la utilización general de CPU del clúster fue inferior al 45 % durante la última semana.

- Para obtener más información, consulte Búsqueda de segmentos concurrentes versión 2.17.

A partir de la versión 2.13 de OpenSearch, puede utilizar la búsqueda de segmentos simultánea para ayudarle a buscar segmentos en paralelo durante la fase de consulta. Para obtener la documentación completa sobre la búsqueda simultánea de segmentos, consulte Concurrent segment search en la documentación de código abierto de OpenSearch. Para obtener información sobre las métricas de Amazon CloudWatch relacionadas con la búsqueda simultánea de segmentos, consulte Instance metrics y UltraWarm metrics.

Existen algunas limitaciones adicionales que se aplican al utilizar la búsqueda de segmentos actual con Amazon OpenSearch Service:

- No puede habilitar la búsqueda de segmentos simultánea en índices en OpenSearch Service.
- De forma predeterminada, OpenSearch Service utiliza un recuento de 2 sectores con el mecanismo de recuento máximo de sectores.

Generación de consultas en lenguaje natural en Amazon OpenSearch Service

La característica de generación de consultas en lenguaje natural de Amazon OpenSearch Service le permite consultar los datos del registro operativo y de seguridad mediante lenguaje natural. OpenSearch es una opción ideal para explorar los datos de registros porque es un motor de análisis y búsqueda de registros altamente escalable y eficiente, y ahora puede utilizar el lenguaje natural para explorar estos registros. Esta característica le permite identificar problemas sin depender del lenguaje de procesamiento canalizado (PPL) de OpenSearch ni tener que buscar definiciones de datos al crear sus consultas. Puede utilizar la característica de generación de consultas en lenguaje natural en los dominios de OpenSearch Service con la versión 2.13 y posteriores. Debe tener el control de acceso detallado habilitado.

Esta característica se creó con OpenSearch Assistant Toolkit. Si desea crear características similares que se conecten a sus modelos lingüísticos de gran tamaño, puede utilizar el kit de herramientas para configurar sus propios agentes y herramientas.

Requisitos previos

Antes de poder utilizar la característica de generación de consultas en lenguaje natural, su dominio debe tener lo siguiente:

- Versión 2.13 o posterior.
- Software del servicio R20240520-P4 o superior.
- Control de acceso detallado habilitado. Para obtener más información, consulte the section called "Habilitar el control de acceso detallado".

Introducción

La generación de consultas en lenguaje natural está habilitada de forma predeterminada en todos los dominios creados con la versión 2.13 o posterior que tengan el control de acceso detallado habilitado.

Para otros dominios, habilítelo seleccionando Habilitar generación de consulta en lenguaje natural y características de Amazon Q Developer.

Después de habilitarlo, navegue a la página Registros en OpenSearch Dashboards. Seleccione Explorador de eventos y haga una pregunta con el asistente para consultas.

Configuración de permisos

Si habilita la generación de consultas en lenguaje natural en un dominio de OpenSearch Service preexistente, es posible que el rol query_assistant_access no esté definido en el dominio. Los usuarios que no sean administradores deben estar asignados a este rol para poder administrar índices templados en los dominios mediante un control de acceso detallado. Para crear el rol query_assistant_access de forma manual, siga estos pasos:

1. En OpenSearch Dashboards, vaya a Seguridad y elija Roles.
2. Elija Crear rol y configure los siguientes permisos de clúster:
   - `cluster:admin/opensearch/ml/config/get`
   - `cluster:admin/opensearch/ml/execute`
   - `cluster:admin/opensearch/ml/predict`
   - `cluster:admin/opensearch/ppl`
3. Asigne al rol el nombre query_assistant_access.
4. Seleccione Crear rol. El rol query_assistant_access ya está disponible.

> **Note**
>
> También debe tener los permisos de índice `indices:admin/mappings/get` y `read` para los índices con los que desee utilizar las preguntas en lenguaje natural.

Automatización de la configuración

Flow Framework es un complemento de OpenSearch que proporciona una forma de automatizar las configuraciones de OpenSearch para casos de uso como la generación de consultas y el chat conversacional. Como el complemento rastrea los recursos que permiten la característica de generación de consultas en lenguaje natural, el índice de Flow Framework almacena una plantilla para cada dominio que utiliza el asistente para consultas.

Flow Framework le permite seleccionar entre un conjunto de plantillas predefinidas o crear sus propias automatizaciones para los conectores, las herramientas, los agentes y otros componentes del machine learning que preparan a OpenSearch como backend para los modelos generativos.

Búsqueda vectorial

La búsqueda vectorial en Amazon OpenSearch Service le permite buscar contenido semánticamente similar mediante incrustaciones de machine learning en lugar de la coincidencia de palabras clave tradicional. La búsqueda vectorial convierte sus datos (texto, imágenes, audio, etc.) en vectores numéricos de alta dimensión (incrustaciones) que capturan el significado semántico del contenido. Al realizar una búsqueda, OpenSearch compara la representación vectorial de la consulta con los vectores almacenados para encontrar los elementos más similares.

La búsqueda vectorial incluye los siguientes componentes clave.

Campos vectoriales

OpenSearch admite el tipo de campo `knn_vector` para almacenar vectores densos con dimensiones configurables (hasta 16.000).

Métodos de búsqueda

- k-NN (k-vecinos más cercanos): encuentra los k vectores más similares
- K-nN aproximado: utiliza algoritmos como HNSW (Hierarchical Navigable Small World) para realizar búsquedas más rápidas en conjuntos de datos de gran tamaño

Métricas de distancia

Admite varios cálculos de similitud, entre los que se incluyen:

- Distancia euclidiana
- Similitud coseno
- Producto Dot

Casos de uso comunes

La búsqueda vectorial admite los siguientes casos de uso habituales.

- Búsqueda semántica: busca documentos con un significado similar, no solo con palabras clave coincidentes
- Sistemas de recomendación: sugiera productos, contenido o usuarios similares
- Búsqueda de imágenes: busca imágenes visualmente similares
- Detección de anomalías: identifique valores atípicos en los patrones de datos
- RAG (Retrieval Augmented Generation): mejore las respuestas de LLM con un contexto relevante

Integración con machine learning

OpenSearch se integra con los siguientes servicios y modelos de machine learning:

- Amazon Bedrock: para generar incrustaciones mediante modelos de base
- Amazon SageMaker AI: para la implementación de modelos de ML personalizados
- Modelos Hugging Face: modelos de incrustación previamente entrenados
- Modelos personalizados: sus propios modelos de incrustación entrenados

La búsqueda vectorial le permite crear aplicaciones sofisticadas impulsadas por la IA que comprenden el contexto y el significado y van mucho más allá de las capacidades tradicionales de coincidencia de texto.

(Versión preliminar) Integración de OpenSearch Service con Amazon S3 Vectors

Amazon OpenSearch Service se integra con Amazon S3 Vectors de las siguientes maneras:

- (Versión preliminar) Importación desde Amazon S3 Vectors a OpenSearch sin servidor
- (Vista previa) Capacidades de búsqueda avanzada con un motor de Amazon S3 Vectors

> **Important**
>
> La integración de Amazon S3 Vectors con OpenSearch Service se encuentra en versión preliminar y está sujeta a cambios.

(Versión preliminar) Importación desde Amazon S3 Vectors a OpenSearch sin servidor

> **Important**
>
> La integración de Amazon S3 Vectors con OpenSearch Service se encuentra en versión preliminar y está sujeta a cambios.

Amazon S3 Vectors ofrece el primer almacén de objetos en la nube con compatibilidad nativa para almacenar y consultar vectores. S3 Vectors proporciona un almacenamiento vectorial rentable, elástico y duradero que se puede consultar en función del significado y la similitud semánticos. Ofrece tiempos de respuesta a las consultas inferiores a un segundo y reduce los costos de carga, almacenamiento y consulta de vectores hasta un 90 %.

Amazon S3 Vectors presenta buckets vectoriales S3, que puede utilizar para almacenar y consultar datos vectoriales, así como acceder a ellos, sin aprovisionar ninguna infraestructura. En un bucket vectorial, puede organizar los datos vectoriales en índices vectoriales. Su bucket vectorial puede tener varios índices vectoriales y cada índice vectorial puede contener millones de vectores. Para obtener más información, consulte Cómo trabajar con Amazon S3 Vectors y buckets de vectores en la Guía del usuario de Amazon S3.

Cada vector se compone de los siguientes elementos:

- Clave única
- Datos vectoriales
- Metadatos opcionales en formato JSON

Los índices vectoriales admiten las funciones de distancia euclidiana y cosenoidal para las operaciones de búsqueda de similitudes.

> **Note**
>
> La principal ventaja de los buckets de vectores es su capacidad para almacenar conjuntos de datos masivos a un costo extremadamente bajo y, al mismo tiempo, proporcionar acceso directo a la API para las operaciones con vectores.

Para obtener más información sobre los buckets de Amazon S3 Vectors, incluido cómo crear uno, consulte Uso de vectores y buckets de Amazon S3 Vectors en la Guía del usuario de Amazon S3. Para obtener más información sobre la integración con OpenSearch Service además de lo que se describe en este tema, consulte Uso de vectores de S3 con OpenSearch Service

Puede usar S3 Vectors con Amazon OpenSearch Service para reducir el costo del almacenamiento vectorial cuando las consultas son menos frecuentes y, después, mover rápidamente esos conjuntos de datos a OpenSearch a medida que aumente la demanda o para mejorar las capacidades de búsqueda.

OpenSearch Service se integra con Amazon S3 Vectors para ofrecer un rendimiento y una funcionalidad mejorados que van más allá de lo que ofrecen los buckets Amazon S3 Vectors por sí solos. Considere esta integración cuando necesite:

- Consultas de mayor rendimiento
- Latencia de búsqueda de menos de un segundo
- Capacidades de análisis avanzadas, como las agregaciones
- Búsqueda híbrida que combina datos de texto y vectores

Esta integración es particularmente útil cuando varias aplicaciones consumen los mismos datos de vectores con diferentes requisitos de rendimiento. Puede hacer que algunas aplicaciones interactúen directamente con los buckets de Amazon S3 Vectors para casos de uso sensibles a los costos, mientras que otras aprovechan la integración de OpenSearch para operaciones de rendimiento crítico.

Arquitecturas de integración

La integración utiliza Amazon OpenSearch Ingestion (OSI) como canalización de datos entre los índices vectoriales de Amazon S3 y las colecciones vectoriales de Amazon OpenSearch sin servidor. OpenSearch Ingestion exporta automáticamente los datos vectoriales del índice vectorial especificado y los ingiere en las colecciones vectoriales de OpenSearch sin servidor para realizar operaciones de búsqueda de alto rendimiento.

> **Note**
>
> Tras la exportación, los datos siguen presentes en el índice vectorial S3. Tiene dos copias de los datos.

Cada índice vectorial se asigna a un índice correspondiente de la colección de OpenSearch Service. En la integración:

- Conserva las dimensiones vectoriales
- Conserva los metadatos
- Optimice la estructura de datos para las capacidades de búsqueda vectorial de OpenSearch

Tras la configuración, OpenSearch Ingestion comienza el proceso de exportación de datos consumiendo los vectores del índice vectorial especificado mediante la API ListVectors de Amazon S3. El servicio procesa los vectores en paralelo para optimizar la velocidad de ingesta y, al mismo tiempo, respeta los límites de escalado de OpenSearch Ingestion y Amazon OpenSearch sin servidor.

Durante la ingesta, el servicio hace lo siguiente:

- Transforma los datos de vectores para que coincidan con el formato esperado para OpenSearch Service
- Conserva la información esencial, incluidos los valores de vectores, los metadatos y las métricas de distancia
- Maneja los escenarios de fallo mediante mecanismos de reintento inteligentes
- Coloca los registros problemáticos en un bucket de Amazon S3 que se utiliza como cola de mensajes fallidos para su posterior análisis

La integración gestiona conjuntos de datos masivos de manera eficiente y el rendimiento depende de las dimensiones de vectores, el tamaño del conjunto de datos y los límites de escalado configurados. OSI puede escalar verticalmente hasta 16 trabajadores por canalización, mientras que OpenSearch sin servidor ajusta automáticamente la capacidad en función de las demandas de ingesta. De forma predeterminada, OpenSearch aumenta a 100 la unidad computacional (OCU) `maxSearch` en OpenSearch sin servidor.

> **Note**
>
> La integración prioriza la rentabilidad mediante:
>
> - Cierre automático de la canalización una vez finalizada la exportación
> - Colección escalable OpenSearch sin servidor
> - Modelo de recursos de pago por uso

Permisos de IAM necesarios

La integración requiere una configuración cuidadosa de los permisos de IAM para permitir una comunicación segura entre los servicios. OpenSearch Ingestion necesita permisos para leer los

índices vectoriales de Amazon S3, escribir en las colecciones vectoriales de OpenSearch Service y administrar las políticas de seguridad asociadas.

Al habilitar la integración mediante el procedimiento que se describe más adelante en este tema, puede elegir una de las siguientes opciones para la administración de permisos:

- Permita que el sistema cree automáticamente un rol de servicio con los permisos necesarios
- Proporcione un rol existente que cumpla con los requisitos

El rol creado automáticamente incluye políticas para:

- Acceso a las API de índices de Amazon S3 Vectors
- Administrar las operaciones de recopilación de OpenSearch Service
- Administración de las operaciones de cola de mensajes fallidos en caso de intentos de ingesta fallidos

Si decide especificar un rol existente, compruebe que dicho rol tenga los siguientes permisos de IAM:

(Obligatorio): permisos de canalización de datos entre OpenSearch Ingestion y OpenSearch sin servidor

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "allowAPIs",
            "Effect": "Allow",
            "Action": [ "aoss:APIAccessAll", "aoss:BatchGetCollection" ],
            "Resource": [ "arn:aws:aoss:*:111122223333:collection/collection-
id" ]
        },
        {
            "Sid": "allowSecurityPolicy",
            "Effect": "Allow",
            "Action": [
                "aoss:CreateSecurityPolicy",
                "aoss:UpdateSecurityPolicy",
```

```
                "aoss:GetSecurityPolicy"
            ],
            "Resource": "*",
            "Condition":{
               "StringLike":{
                  "aoss:collection": [ "collection-name" ]
                },
               "StringEquals": {
                  "aws:ResourceAccount": [ "111122223333" ]
               }
            }
        }
    ]
}
```

(Obligatorio): permisos de ingesta de datos entre OpenSearch Ingestion y Amazon S3 (cola de mensajes fallidos)

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "s3Access",
            "Effect": "Allow",
            "Action": [
              "s3:PutObject"
            ],
            "Resource": [ "arn:aws:s3:::bucket/*" ]
        }
    ]
}
```

(Obligatorio): permisos de ingesta de datos entre OpenSearch Ingestion y Amazon S3 Vectors

JSON

```
{
```

```
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AllowS3VectorIndexAccess",
            "Effect": "Allow",
            "Action": [
               "s3vectors:ListVectors",
               "s3vectors:GetVectors"
            ],
            "Resource": [
                "arn:aws:s3vectors:us-east-1:111122223333:bucket/bucket-name/
index/index-name"
            ]
        }
    ]
}
```

(Obligatorio si el cifrado AWS KMS está activado): permisos de descifrado para la comunicación entre OpenSearch Ingestion y Amazon S3 Vectors

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "allowS3VectorDecryptionOfCustomManagedKey",
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": [ "arn:aws:kms:us-east-1:111122223333:key/key-id" ],
            "Condition": {
                "StringEquals": {
                    "kms:ViaService": "s3vectors.us-east-1.amazonaws.com",
                    "kms:EncryptionContext:aws:s3vectors:arn":
 "arn:aws:s3vectors:us-east-1:111122223333:bucket/example-bucket"
                 }
             }
        }
    ]
```

```
}
```

Configuración de la integración de Amazon S3 Vectors con OpenSearch

Utilice el siguiente procedimiento para configurar la integración de Amazon S3 Vectors con OpenSearch sin servidor.

> **Note**
>
> Si ha iniciado el proceso de configuración de la integración desde la consola de Amazon S3 seleccionando la opción Exportar a OpenSearch en la página de Bucket de vector, algunos de los pasos del siguiente procedimiento no son aplicables, como se indica en el procedimiento.

Para configurar la integración de Amazon S3 Vectors con OpenSearch sin servidor

1. Abra la página Importar el índice vectorial de S3 al motor vectorial OpenSearch en la consola de Amazon OpenSearch Service. La página se muestra automáticamente si ha hecho clic en Exportar a OpenSearch en la consola de Amazon S3. Si está comenzando en la consola de OpenSearch, seleccione Integración en el menú de navegación de la izquierda y, a continuación, seleccione Importar índice vectorial de S3.
2. En la sección Source, si comenzó en la consola de Amazon S3, compruebe que el nombre del índice vectorial y su nombre de recurso de Amazon (ARN) ya estén especificados. Si comenzó en la consola OpenSearch, introduzca el ARN del índice en el campo ARN del índice vectorial S3.
3. En la sección Agregar cuentas elija una opción. Si elige un rol existente, verifique que tenga todos los permisos necesarios para la integración, como se describe en Permisos de IAM necesarios.
4. (Opcional) Amplíe Configuración adicional. Para Habilitar la redundancia (réplicas activas), recomendamos dejar esta opción seleccionada para los entornos de producción. Al crear su primera colección, OpenSearch sin servidor crea una instancia de dos OCU: una para la indexación y otra para la búsqueda. Para garantizar la alta disponibilidad, también lanza un conjunto de nodos en espera en otra zona de disponibilidad. Para fines de desarrollo y pruebas, puede deshabilitar la configuración Habilitar redundancia para una colección, lo que elimina las dos réplicas en espera y solo crea una instancia de dos OCU. De forma predeterminada, las

réplicas activas redundantes están habilitadas, lo que significa que se crean instancias para un total de cuatro OCU para la primera colección de una cuenta.

En Agregar clave de AWS KMS administrada por el cliente para el vector de Amazon OpenSearch sin servidor, elija esta opción para cifrar los datos de la colección vectorial mediante una clave administrada por el cliente. De forma predeterminada, OpenSearch usa una Clave administrada de AWS.

5. Si ha iniciado este proceso haciendo clic en la opción Exportar a OpenSearch en la consola de Amazon S3, en la sección Detalles de exportación se enumeran los pasos que OpenSearch realizará a continuación. Cuando esté listo para continuar, seleccione Exportar.

   Si inició este proceso en la consola de OpenSearch Service, en la sección Detalles de importación se enumeran los pasos que OpenSearch realizará a continuación. Cuando esté listo para continuar, elija Importar.

   OpenSearch abre la página de historial para mostrar todas las exportaciones/importaciones de índices vectoriales de Amazon S3 a índices de OpenSearch sin servidor.

Tras una ingesta correcta, OSI detiene automáticamente la canalización para evitar costos innecesarios y, al mismo tiempo, mantener los datos exportados en OpenSearch. Puede supervisar el progreso de la integración mediante las métricas de CloudWatch y acceder a registros detallados para solucionar problemas.

La colección OpenSearch permanece activa y disponible para consultas una vez completada la ingesta inicial. Puede realizar:

- Búsqueda de similitud
- Agregaciones
- Análisis operativo

(Vista previa) Capacidades de búsqueda avanzada con un motor de Amazon S3 Vectors

> **Important**
>
> La integración de Amazon S3 Vectors con OpenSearch Service se encuentra en versión preliminar y está sujeta a cambios.

Amazon OpenSearch Service ofrece la posibilidad de utilizar Amazon S3 como motor vectorial para índices vectoriales. Esta característica le permite descargar datos vectoriales a Amazon S3 y, al mismo tiempo, mantener las capacidades de búsqueda vectorial en menos de un segundo a un bajo costo.

Con esta característica, OpenSearch almacena las incrustaciones vectoriales en un índice de Amazon S3 Vectors y, al mismo tiempo, mantiene otros campos del documento en el almacenamiento del clúster de OpenSearch. Este método ofrece las siguientes ventajas:

- Durabilidad: Los datos escritos en S3 Vectors se almacenan en S3, que se ha diseñado para ofrecer una durabilidad de datos del 99,999999999 % (11 nueves).
- Escalabilidad: transfiera grandes conjuntos de datos vectoriales a S3 sin consumir almacenamiento en clústeres.
- Rentabilidad: optimice los costos de almacenamiento para cargas de trabajo con gran cantidad de vectores.

OpenSearch tiene los siguientes requisitos para usar índices vectoriales S3:

- Un dominio de OpenSearch con versión 2.19 o posterior
- Instancias optimizadas para OpenSearch
- Última versión del parche para su versión de OpenSearch

Activación de S3 Vectors

Al crear un dominio nuevo o actualizar un dominio existente, puede elegir la opción Habilitar S3 Vectors como motor en la sección Características avanzadas. Esta configuración le permite a

OpenSearch crear un bucket vectorial de S3 al utilizar S3 Vectors como motor. Al activar esta opción, OpenSearch configura S3 Vectors para su dominio de la siguiente manera:

1. Al crear dos nuevas concesiones en la clave AWS KMS configurada con su dominio:
   - Una concesión para las tareas de indexación en segundo plano de S3 Vectors con privilegios de descifrado
   - Una subvención para que OpenSearch cree buckets de vectores de S3 con permisos `GenerateDataKey`
2. Configurar la clave KMS utilizada por su dominio de OpenSearch como CMK para cifrar el resto de los datos del índice vectorial.

Creación de índices con el motor vectorial de S3

Después de configurar un dominio, puede crear uno o más índices k-NN con campos, con el uso de `s3vector` como motor vectorial de backend en las asignaciones de índices. Puede configurar diferentes campos vectoriales con diferentes tipos de motores en función de su caso de uso.

> **Important**
>
> Solo puede utilizar el motor `s3vector` para asignar una definición de campo durante la creación del índice. No puede agregar ni actualizar la asignación con el motor `s3vector` después de crear el índice.

Estos son algunos ejemplos que crean índices de motores vectoriales de S3.

Ejemplo: creación de un índice k-NN con el motor vectorial de S3

```
PUT my-first-s3vector-index
{
  "settings": {
    "index": {
      "knn": true
    }
  },
  "mappings": {
    "properties": {
        "my_vector_1": {
          "type": "knn_vector",
```

```
          "dimension": 2,
          "space_type": "l2",
          "method": {
            "engine": "s3vector"
          }
        },
        "price": {
          "type": "float"
        }
    }
  }
}
```

Ejemplo: creación de un índice k-NN con motores vectoriales de S3 y FAISS

Este ejemplo destaca el hecho de que se pueden utilizar varios motores vectoriales dentro del mismo índice.

```
PUT my-vector-index
{
  "settings": {
    "index": {
      "knn": true
    }
  },
  "mappings": {
    "properties": {
        "my_vector_1": {
          "type": "knn_vector",
          "dimension": 2,
          "space_type": "l2",
          "method": {
            "engine": "s3vector"
          }
        },
        "price": {
          "type": "float"
        },
        "my_vector_2": {
            "type": "knn_vector",
            "dimension": 2,
            "space_type": "cosine",
            "method": {
```

```
                "name": "hnsw",
                "engine": "faiss",
                "parameters": {
                    "ef_construction": 128,
                    "m": 24
                }
            }
        }
    }
  }
}
```

Ejemplo no compatible: agregar el motor vectorial de S3 tras la creación del índice

El siguiente enfoque no es compatible y fallará.

```
PUT my-first-s3vector-index
{
  "settings": {
    "index": {
      "knn": true
    }
  }
}

PUT my-first-s3vector-index/_mapping
{
  "properties": {
        "my_vector_1": {
          "type": "knn_vector",
          "dimension": 2,
          "space_type": "l2",
          "method": {
            "engine": "s3vector"
          }
        },
        "price": {
          "type": "float"
        }
    }
}
```

Limitaciones de funcionalidad

Tenga en cuenta las siguientes limitaciones antes de utilizar el motor `s3vector` en un índice:

Características y comportamientos no compatibles con el motor s3vector

Característica	Comportamiento
Índice de división, contracción y clonación	Estas API fallan cuando se utilizan con un índice configurado con el motor `s3vector` en el campo `knn_vector` .
Instantáneas	Los índices que utilizan el motor `s3vector` no admiten instantáneas. Para dominios administrados: • Las instantáneas automatizadas solo incluyen índices que no utilizan el motor `s3vector`. • Las solicitudes manuales de instantáneas para los índices `s3vector` fallan. **Note** Si bien las instantáneas no son compatibles con la recuperación en un momento dado, el motor `s3vector`, junto con las instancias optimizadas para OpenSearch, ofrecen una durabilidad de 11 nueves.
Nivel UltraWarm	Los índices configurados con el motor `s3vector` no pueden migrar al nivel UltraWarm.
Replicación entre clústeres	Los índices configurados con el motor `s3vector` no admiten la replicación entre clústeres.
Protección contra eliminaciones accidentales	Como las instantáneas no son compatibles con los índices que utilizan el motor `s3vector`, la protección contra la eliminación accidental no está disponible. Aún puedes restaurar otros índices en el dominio.

Característica	Comportamiento
Búsqueda radial	Las consultas con búsqueda radial no se admiten en los campos que utilizan el motor `s3vector`.

Indexación de documentos

Tras crear un índice con el motor vectorial de S3, puede ingerir documentos mediante la API estándar `_bulk`. OpenSearch transfiere automáticamente los datos vectoriales de los campos `knn_vector` que utilizan el motor `s3vector` al índice vectorial de S3 en tiempo real. OpenSearch conservará los datos que pertenezcan a otros campos `knn_vector` o a campos que utilicen diferentes motores en su propia capa de almacenamiento.

Para todas las solicitudes masivas que se acepten, OpenSearch garantiza que todos los datos (vectoriales y no vectoriales) sean duraderos. Si una solicitud recibe un acuse de recibo negativo, no se garantiza la durabilidad de los documentos incluidos en esa solicitud masiva. Debería volver a intentar dichas solicitudes.

Ejemplo de indexación masiva

```
POST _bulk
{ "index": { "_index": "my-first-s3vector-index", "_id": "1" } }
{ "my_vector_1": [1.5, 2.5], "price": 12.2 }
{ "index": { "_index": "my-first-s3vector-index", "_id": "2" } }
{ "my_vector_1": [2.5, 3.5], "price": 7.1 }
{ "index": { "_index": "my-first-s3vector-index", "_id": "3" } }
{ "my_vector_1": [3.5, 4.5], "price": 12.9 }
{ "index": { "_index": "my-first-s3vector-index", "_id": "4" } }
{ "my_vector_1": [5.5, 6.5], "price": 1.2 }
{ "index": { "_index": "my-first-s3vector-index", "_id": "5" } }
{ "my_vector_1": [4.5, 5.5], "price": 3.7 }
```

Buscar documentos

Puede buscar en su índice mediante la API `_search` estándar para ejecutar consultas de texto, k-NN o híbridas. En el caso de las consultas en campos `knn_vector` configurados con el motor `s3vector`, OpenSearch transfiere automáticamente la consulta al índice vectorial de S3 correspondiente.

> **Note**
>
> Con el motor `s3vector`, la semántica de actualización solo se aplica a los campos que no utilizan el motor `s3vector`. Sin embargo, los datos vectoriales descargados en `s3vector` estarán visibles inmediatamente después de que un documento se indexe correctamente.

Consulta de búsqueda de ejemplo

```
GET my-first-s3vector-index/_search
{
  "size": 2,
  "query": {
    "knn": {
      "my_vector_1": {
        "vector": [2.5, 3.5],
        "k": 2
      }
    }
  }
}
```

Parámetros de asignación compatibles

Con el motor `s3vector`, el campo `knn_vector` admite los siguientes parámetros en las asignaciones.

Parámetros de campo vectorial

Parámetro	Obligatorio	Descripción	Valores admitidos
`type`	Sí	El tipo de campo presente en el documento.	`knn_vector`
`dimension`	Sí	La dimensión de cada vector que se ingerirá en el índice.	>0, <=4096
`space_type`	No	El espacio vectorial utilizado para calcular la distancia entre los vectores.	`l2`, `cosinesimil`

Parámetro	Obligatorio	Descripción	Valores admitidos
`method.engine`	Sí	El motor k-NN aproximado que se utilizará para la indexación y la búsqueda.	`s3vector`
`method.name`	No	El método del vecino más cercano	""

> **Important**
>
> El motor `knn_vector` no admite los tipos de campos anidados de `s3vector`

Facturación y medición

Hasta que se anuncie la medición, esta característica no se facturará.

Deshabilitación del motor s3vector

Antes de deshabilitar el motor `s3vector`, elimine todos los índices que lo utilizan actualmente. Si no lo hace, cualquier intento de deshabilitar el motor fallará.

Tenga en cuenta también que al habilitar o deshabilitar el motor `s3vector` se desencadena una implementación azul/verde en su dominio.

Para deshabilitar el motor `s3vector`, edite la configuración de su dominio y configure `S3VectorsEngine.Enabled: false`.

Búsqueda k-Nearest Neighbor (k-NN) en Amazon OpenSearch Service

Con la sigla en inglés de su algoritmo asociado vecinos más cercanos de k, k-NN para Amazon OpenSearch Service permite buscar puntos en un espacio vectorial y encontrar los “vecinos más cercanos” de esos puntos por distancia euclidiana o similitud de coseno. Los casos de uso incluyen recomendaciones (por ejemplo, una característica de “otras canciones que podrían gustarte” en una aplicación de música), reconocimiento de imágenes y detección de fraude.

> **Note**
>
> Esta documentación proporciona una breve descripción del complemento k-NN, así como las limitaciones al usar el complemento con OpenSearch Service administrado. Para obtener la documentación completa del complemento k-NN, que incluye ejemplos simples y complejos, referencias de parámetros y la referencia completa de la API del complemento, consulte la documentación de OpenSearch de código abierto. La documentación de código abierto también cubre el ajuste del rendimiento y la configuración de clústeres específica de K-NN.

Introducción a k-NN

Para utilizar k-NN, debe crear un índice con la configuración `index.knn` y agregar uno o más campos del tipo de datos `knn_vector`.

```
PUT my-index

{
  "settings": {
    "index.knn": true
  },
  "mappings": {
    "properties": {
      "my_vector1": {
        "type": "knn_vector",
        "dimension": 2
      },
      "my_vector2": {
        "type": "knn_vector",
        "dimension": 4
      }
    }
  }
}
```

El tipo de datos `knn_vector` admite una sola lista de hasta 10 000 flotadores, con el número de flotadores definido por el parámetro `dimension` requerido. Después de crear el índice, agregue algunos datos.

```
POST _bulk
```

```
{ "index": { "_index": "my-index", "_id": "1" } }
{ "my_vector1": [1.5, 2.5], "price": 12.2 }
{ "index": { "_index": "my-index", "_id": "2" } }
{ "my_vector1": [2.5, 3.5], "price": 7.1 }
{ "index": { "_index": "my-index", "_id": "3" } }
{ "my_vector1": [3.5, 4.5], "price": 12.9 }
{ "index": { "_index": "my-index", "_id": "4" } }
{ "my_vector1": [5.5, 6.5], "price": 1.2 }
{ "index": { "_index": "my-index", "_id": "5" } }
{ "my_vector1": [4.5, 5.5], "price": 3.7 }
{ "index": { "_index": "my-index", "_id": "6" } }
{ "my_vector2": [1.5, 5.5, 4.5, 6.4], "price": 10.3 }
{ "index": { "_index": "my-index", "_id": "7" } }
{ "my_vector2": [2.5, 3.5, 5.6, 6.7], "price": 5.5 }
{ "index": { "_index": "my-index", "_id": "8" } }
{ "my_vector2": [4.5, 5.5, 6.7, 3.7], "price": 4.4 }
{ "index": { "_index": "my-index", "_id": "9" } }
{ "my_vector2": [1.5, 5.5, 4.5, 6.4], "price": 8.9 }
```

A continuación, puede buscar los datos mediante el tipo de consulta `knn`.

```
GET my-index/_search
{
  "size": 2,
  "query": {
    "knn": {
      "my_vector2": {
        "vector": [2, 3, 5, 6],
        "k": 2
      }
    }
  }
}
```

En este caso, `k` es el número de vecinos que desea que devuelva la consulta, pero también debe incluir la opción `size`. De lo contrario, obtendrá `k` resultados para cada partición (y cada segmento) en lugar de `k` resultados para toda la consulta. k-NN admite un valor de `k` máximo de 10 000.

Si mezcla la consulta `knn` con otras cláusulas, es posible que reciba menos resultados `k`. En este ejemplo, la cláusula `post_filter` reduce el número de resultados de 2 a 1.

```
GET my-index/_search

{
  "size": 2,
  "query": {
    "knn": {
      "my_vector2": {
        "vector": [2, 3, 5, 6],
        "k": 2
      }
    }
  },
  "post_filter": {
    "range": {
      "price": {
        "gte": 6,
        "lte": 10
      }
    }
  }
}
```

Si necesita gestionar un gran volumen de consultas y, al mismo tiempo, mantener un rendimiento óptimo, puede utilizar la API `_msearch` para crear una búsqueda masiva con JSON y enviar una única solicitud para realizar varias búsquedas:

```
GET _msearch

{ "index": "my-index"}
{ "query": { "knn": {"my_vector2":{"vector": [2, 3, 5, 6],"k":2 }} } }
{ "index": "my-index", "search_type": "dfs_query_then_fetch"}
{ "query": { "knn": {"my_vector1":{"vector": [2, 3],"k":2 }} } }
```

En el siguiente video se muestra cómo configurar búsquedas vectoriales masivas para consultas K-NN.

Diferencias, ajuste y limitaciones de k-NN

OpenSearch permite modificar todas las configuraciones de k-NN mediante la API `_cluster/settings`. En OpenSearch Service, puede cambiar todas las configuraciones, excepto

`knn.memory.circuit_breaker.enabled` y `knn.circuit_breaker.triggered`. Las estadísticas k-NN se incluyen como Métricas de Amazon CloudWatch.

En particular, verifique la métrica `KNNGraphMemoryUsage` en cada nodo de datos con respecto a la estadística `knn.memory.circuit_breaker.limit` y la RAM disponible para el tipo de instancia. OpenSearch Service utiliza la mitad de la RAM de una instancia para la pila de Java (hasta un tamaño de pila de 32 GiB). De forma predeterminada, k-NN utiliza hasta el 50 % de la mitad restante, por lo que un tipo de instancia con 32 GiB de RAM puede acomodar 8 GiB de gráficos (32 * 0,5 * 0,5). El rendimiento puede verse afectado si el uso de la memoria gráfica supera este valor.

Puede migrar un índice k-NN creado en la versión 2.x o posterior a UltraWarm o almacenamiento en frío en un dominio con la versión 2.17 o posterior.

La API de borrado de caché y las API de calentamiento para los índices k-NN están bloqueadas para los índices cálidos. Cuando se inicia la primera consulta para el índice, descarga los archivos de gráficos de Amazon S3 y carga el gráfico en la memoria. Del mismo modo, cuando el TTL de los gráficos caduca, los archivos se eliminan automáticamente de la memoria.

Utilización de OpenSearch Dashboards con Amazon OpenSearch Service

OpenSearch Dashboards es una herramienta popular de visualización de código abierto diseñada para trabajar con OpenSearch. Amazon OpenSearch Service proporciona una instalación de OpenSearch Dashboards con todos los dominios de OpenSearch Service. OpenSearch Dashboards se ejecuta en los nodos de datos más importantes del dominio.

OpenSearch Dashboards es una herramienta de visualización para explorar y analizar datos dentro de un único dominio de OpenSearch. Por el contrario, la interfaz de usuario centralizada de OpenSearch (también denominada aplicación OpenSearch) es una interfaz de usuario basada en la nube que se conecta a varios dominios de OpenSearch, colecciones de OpenSearch sin servidor y orígenes de datos de AWS. Incluye espacios de trabajo para casos de uso específicos, como los análisis de observabilidad y seguridad y proporciona una experiencia unificada en todos los conjuntos de datos. Si bien Dashboards está vinculado a dominios individuales, la interfaz de usuario centralizada permite la integración y el análisis de datos entre dominios. Para obtener más información, consulte OpenSearch UI.

Puede encontrar un enlace a OpenSearch Dashboards en el panel del dominio en la consola de OpenSearch Service. Para los dominios que ejecutan OpenSearch, la URL es *domain-endpoint*`/_dashboards/`. Para los dominios que ejecutan una versión heredada de Elasticsearch, la URL es *domain-endpoint*`/_plugin/kibana`.

Las consultas que utilizan esta instalación de OpenSearch Dashboards predeterminada tienen un tiempo de espera de 300 segundos.

> **Note**
>
> En esta documentación se analiza OpenSearch Dashboards en el contexto de Amazon OpenSearch Service, incluidas las diferentes formas de conectarse a ellos. Para obtener una documentación completa, que incluye una guía de introducción, instrucciones para crear un panel, administración de paneles y el lenguaje de consulta de paneles (DQL), consulte OpenSearch Dashboards en la documentación de OpenSearch de código abierto.

Control del acceso a Dashboards

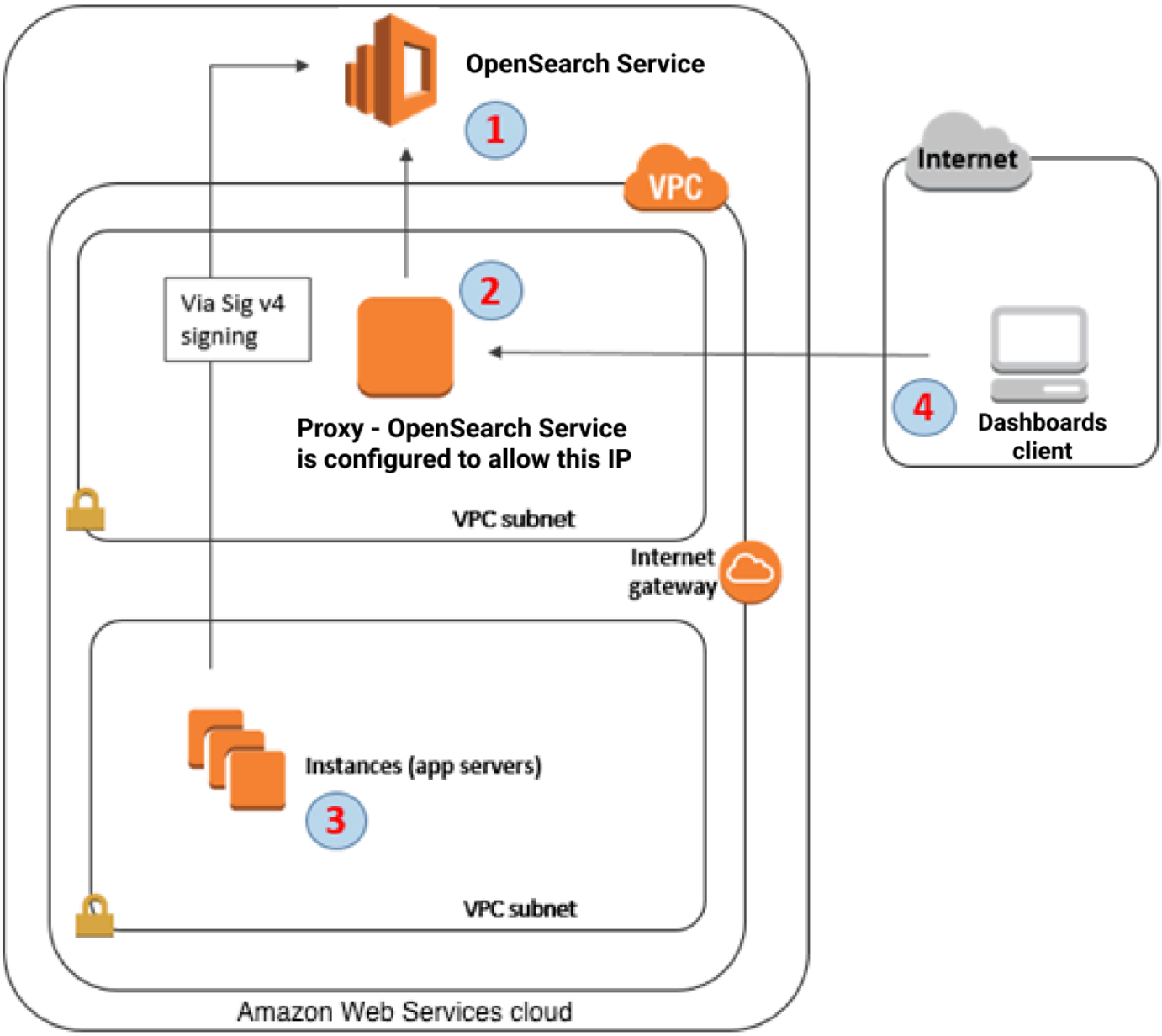
Dashboards no admite de forma nativa los usuarios y roles de IAM, pero OpenSearch Service ofrece varias soluciones para controlar el acceso a Dashboards:

- Habilite SAML authentication for Dashboards (Autenticación SAML para Dashboards).
- Utilice el control de acceso detallado con autenticación básica de HTTP.
- Configure la autenticación de Cognito para Dashboards.
- Para los dominios de acceso público, configure una política de acceso basada en IP que utilice o no un servidor proxy.
- Para los dominios de acceso a la VPC, configure una política de acceso abierto que utilice o no un servidor proxy, y grupos de seguridad para controlar el acceso. Para obtener más información, consulte the section called “Acerca de las políticas de acceso en los dominios de VPC”.

Utilización de un proxy para acceder a OpenSearch Service desde Dashboards

> **Note**
>
> Este proceso solo es de aplicación si el dominio utiliza acceso público y no se desea emplear la autenticación de Cognito. Consulte the section called “Control del acceso a Dashboards ”.

Dado que Dashboards es una aplicación de JavaScript, las solicitudes provienen de la dirección IP del usuario. El control de acceso basado en IP podría resultar poco práctico debido al gran número de direcciones IP a las que sería necesario brindar acceso para que cada usuario tenga acceso a Dashboards. Una alternativa consiste en colocar un servidor proxy entre OpenSearch Dashboards y OpenSearch Service. A continuación, puede agregar una política de acceso basada en IP que permita solicitudes desde solo una única dirección IP, la del proxy. En el siguiente diagrama se muestra esta configuración.



1. Este es el dominio de OpenSearch Service. IAM proporciona acceso autorizado a este dominio. Otra política de acceso basada en IP proporciona acceso al servidor proxy.
2. Este es el servidor proxy que se ejecuta en una instancia de Amazon EC2.
3. Otras aplicaciones pueden utilizar el proceso de firma Signature Version 4 para enviar solicitudes autenticadas a OpenSearch Service.
4. Los clientes de OpenSearch Dashboards se conectan a su dominio de OpenSearch Service a través del proxy.

Para habilitar esta clase de configuración, necesita una política basada en recursos que especifica funciones y direcciones IP. A continuación, mostramos un ejemplo de política:

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Resource":"arn:aws:es:us-west-2:111111111111:domain/my-domain/*",
         "Principal":{
            "AWS":"arn:aws:iam::111111111111:role/allowedrole1"
         },
         "Action":[
            "es:ESHttpGet"
         ],
         "Effect":"Allow"
      },
      {
         "Effect":"Allow",
         "Principal":{
            "AWS":"*"
         },
         "Action":"es:*",
         "Condition":{
            "IpAddress":{
               "aws:SourceIp":[
                  "203.0.113.0/24",
                  "2001:DB8:1234:5678::/64"
               ]
            }
         },
         "Resource":"arn:aws:es:us-west-2:111111111111:domain/my-domain/*"
      }
   ]
}
```

Recomendamos configurar la instancia EC2 mediante la ejecución del el servidor proxy con una dirección IP elástica. De esta forma, puede sustituir la instancia cuando sea necesario y continuar adjuntando la misma dirección IP pública a la misma. Para obtener más información, consulte Direcciones IP elásticas en la Guía del usuario de Amazon EC2.

Si utiliza un servidor proxy y la autenticación de Cognito, es posible que tenga que agregar parámetros de configuración para Dashboards y Amazon Cognito con el fin de evitar errores de tipo `redirect_mismatch`. Consulte el siguiente ejemplo de `nginx.conf`:

```
server {
    listen 443;
    server_name $host;
    rewrite ^/$ https://$host/_plugin/_dashboards redirect;

    ssl_certificate          /etc/nginx/cert.crt;
    ssl_certificate_key      /etc/nginx/cert.key;

    ssl on;
    ssl_session_cache  builtin:1000  shared:SSL:10m;
    ssl_protocols  TLSv1 TLSv1.1 TLSv1.2;
    ssl_ciphers HIGH:!aNULL:!eNULL:!EXPORT:!CAMELLIA:!DES:!MD5:!PSK:!RC4;
    ssl_prefer_server_ciphers on;

    location /_plugin/_dashboards {
        # Forward requests to Dashboards
        proxy_pass https://$dashboards_host/_plugin/_dashboards;

        # Handle redirects to Cognito
        proxy_redirect https://$cognito_host https://$host;

        # Update cookie domain and path
        proxy_cookie_domain $dashboards_host $host;
        proxy_cookie_path / /_plugin/_dashboards/;

        # Response buffer settings
        proxy_buffer_size 128k;
        proxy_buffers 4 256k;
        proxy_busy_buffers_size 256k;
    }

    location ~ \/(log|sign|fav|forgot|change|saml|oauth2) {
        # Forward requests to Cognito
        proxy_pass https://$cognito_host;

        # Handle redirects to Dashboards
        proxy_redirect https://$dashboards_host https://$host;

        # Update cookie domain
```

```
        proxy_cookie_domain $cognito_host $host;
    }
}
```

> **Note**
>
> (Opcional) Si decide aprovisionar un nodo coordinador dedicado, este comenzará a alojar automáticamente OpenSearch Dashboards. Como resultado, aumenta la disponibilidad de los recursos del nodo de datos, como la CPU y la memoria. Esta mayor disponibilidad de los recursos de los nodos de datos puede ayudar a mejorar la resiliencia general de su dominio.

Configuración de OpenSearch Dashboards para utilizar un servidor de mapas de WMS

La instalación predeterminada de OpenSearch Dashboards para OpenSearch Service incluye un servicio de mapas, excepto para los dominios en las regiones de India y China. El servicio de mapas admite hasta 10 niveles de acercamiento/alejamiento.

Con independencia de la región, puede configurar Dashboards para que utilice un servidor de Web Map Service (WMS) diferente para las visualizaciones de mapas de coordenadas. Las visualizaciones de mapas de región solo admiten el servicio de mapas predeterminado.

Para configurar Dashboards para que utilice un servidor de mapas de WMS:

1. Abra Dashboards.
2. Seleccione Stack Management (Gestión de pilas).
3. Seleccione Advanced Settings (Configuración avanzada).
4. Localice visualization:tileMap:WMSdefaults.
5. Cambie `enabled` a `true` y `url` a la URL de un servidor de mapas WMS válido.

```
{
  "enabled": true,
  "url": "wms-server-url",
  "options": {
    "format": "image/png",
    "transparent": true
```

```
  }
}
```

6. Seleccione Save changes (Guardar cambios).

Para aplicar el nuevo valor predeterminado a las visualizaciones, es posible que deba volver a cargar Dashboards. Si guardó visualizaciones, elija Options (Opciones) después de abrir la visualización. Verifique que el Servidor de mapas WMS esté habilitado y que la URL de WMS contenga el servidor de mapas preferido y, a continuación, elija Apply changes (Aplicar cambios).

> **Note**
>
> Los servicios de mapa suelen estar sujetos a restricciones o cuotas de licencias. El usuario es responsable de todas estas consideraciones en cualquier servidor de mapas que especifique. Los servicios de mapa de U.S. Geological Survey podrían resultarle útiles para hacer pruebas.

Conexión de un servidor local de Dashboards a OpenSearch Service

Si invirtió mucho tiempo en configurar su propia instancia de OpenSearch Dashboards, podrá utilizarla en lugar de (o además de) la instancia predeterminada de Dashboards que OpenSearch Service ofrece. El siguiente procedimiento sirve para dominios que utilicen el control de acceso detallado con una política de acceso abierto.

Conexión de un servidor local de Dashboards a OpenSearch Service

1. En el dominio de OpenSearch Service, cree un usuario con los permisos adecuados:
   a. En Dashboards, vaya a Security (Seguridad), Internal users (Usuarios internos) y elija Create internal user (Crear usuario interno).
   b. Proporcione un nombre de usuario y una contraseña y elija Create (Crear).
   c. Vaya a Roles (Roles) y elija un rol.
   d. Seleccione Mapped users (Usuarios asignados) y Seleccione Manage mapping (Gestión de mapas).
   e. En Users (Usuarios), agregue el nombre de usuario y elija Map (Asignar).

2. Descargue e instale la versión apropiada del complemento de seguridad de OpenSearch en la instalación de OSS de Dashboards autoadministrada.
3. En el servidor local de Dashboards, abra el archivo `config/opensearch_dashboards.yml` y agregue el punto de enlace de OpenSearch Service con el nombre de usuario y la contraseña que creó anteriormente:

```
opensearch.hosts: ['https://domain-endpoint']
opensearch.username: 'username'
opensearch.password: 'password'
```

Puede utilizar el siguiente ejemplo de archivo `opensearch_dashboards.yml`:

```
server.host: '0.0.0.0'

opensearch.hosts: ['https://domain-endpoint']

opensearchDashboards.index: ".username"

opensearch.ssl.verificationMode: none # if not using HTTPS

opensearch_security.auth.type: basicauth
opensearch_security.auth.anonymous_auth_enabled: false
opensearch_security.cookie.secure: false # set to true when using HTTPS
opensearch_security.cookie.ttl: 3600000
opensearch_security.session.ttl: 3600000
opensearch_security.session.keepalive: false
opensearch_security.multitenancy.enabled: false
opensearch_security.readonly_mode.roles: ['opensearch_dashboards_read_only']
opensearch_security.auth.unauthenticated_routes: []
opensearch_security.basicauth.login.title: 'Please log in using your username and
 password'

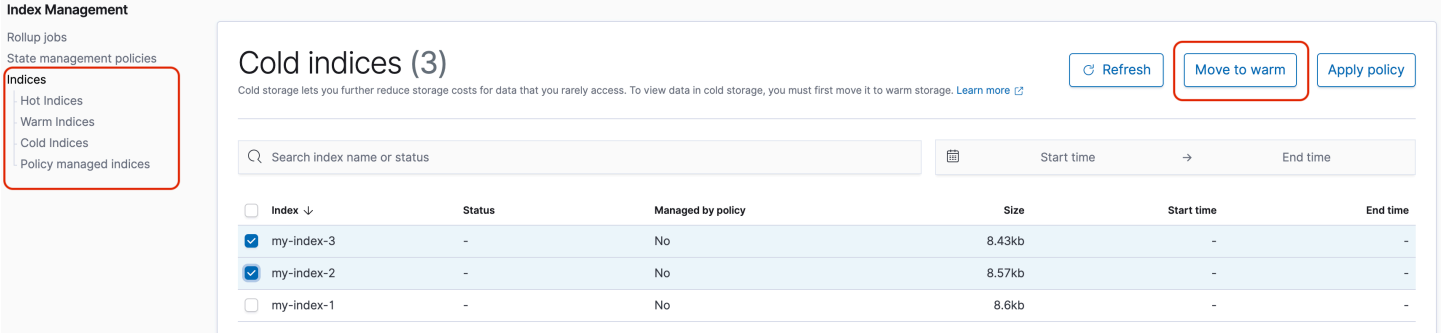
opensearch.username: 'username'
opensearch.password: 'password'
opensearch.requestHeadersWhitelist: [authorization, securitytenant,
 security_tenant]
```

Para ver los índices de OpenSearch Service, inicie el servidor local de Dashboards, vaya a Herramientas para desarrolladores y ejecute el siguiente comando:

```
GET _cat/indices
```

Administración de índices en OpenSearch Dashboards

La instalación de OpenSearch Dashboards en el dominio de OpenSearch Service proporciona una interfaz de usuario útil para administrar índices en diferentes niveles de almacenamiento del dominio. Seleccione Index Management (Administración de índices) desde el menú principal de Dashboards para ver todos los índices en los tipos de almacenamiento en caliente, UltraWarm y en frío, así como los índices administrados por políticas de administración de estados de índices (ISM). Utilice la administración de índices para mover índices entre el almacenamiento en frío y el almacenamiento en caliente y para monitorear las migraciones entre los tres niveles.



Tenga en cuenta que no verá las opciones de índice de calor, templado y frío a menos que tenga habilitado el almacenamiento UltraWarm o en frío.

Características adicionales

La instalación predeterminada de Dashboards en cada dominio de OpenSearch Service cuenta con algunas características adicionales:

- Interfaces de usuario para los distintos complementos de OpenSearch
- Inquilinos
- Informes

  Utilice Reporting (Informes) para generar informes CSV bajo demanda desde la página Discover e informes PDF o PNG de paneles o visualizaciones. Los informes CSV tienen un límite de 10 000 filas.
- Diagramas de Gantt
- Cuadernos

Uso de OpenSearch UI en Amazon OpenSearch Service

OpenSearch UI (interfaz de usuario) es una experiencia de análisis operativo modernizada para Amazon OpenSearch Service que proporciona una vista unificada para que pueda interactuar con los datos de varios orígenes. A diferencia de OpenSearch Dashboards, que solo funciona con el dominio o colección que lo aloja, OpenSearch UI está alojado en la Nube de AWS. Esto permite que OpenSearch UI alcance una alta disponibilidad y siga siendo funcional durante las actualizaciones del clúster, así como que se conecte de forma nativa con varios orígenes de datos. Para obtener más información acerca de OpenSearch Dashboards, consulte OpenSearch Dashboards.

Estas son algunas de las características principales de OpenSearch UI:

- Soporte para varios orígenes de datos: OpenSearch UI puede conectarse con varios orígenes de datos para crear una vista integral. Esto incluye dominios y colecciones sin servidor de OpenSearch, así como orígenes de datos de AWS integrados como Amazon CloudWatch, Amazon Security Lake y Amazon Simple Storage Service (Amazon S3).
- Sin tiempo de inactividad durante las actualizaciones: OpenSearch UI está alojado en la Nube de AWS. Esto significa que OpenSearch permanece operativo y puede recuperar datos de los clústeres durante los procesos de actualización.
- Espacios de trabajo: espacios seleccionados para la colaboración en equipo para diversos flujos de trabajo, como observabilidad, análisis de seguridad y búsqueda. Puede definir la configuración de privacidad y administrar los permisos de los colaboradores en su espacio de trabajo.
- Inicio de sesión único: OpenSearch UI funciona con AWS IAM Identity Center y SAML a través de una federación de AWS Identity and Access Management (IAM) para integrarse con los proveedores de identidad y crear una experiencia de inicio de sesión único para sus usuarios finales.
- Análisis impulsado por IA generativa: OpenSearch UI admite la generación de consultas en lenguaje natural para ayudar a generar las consultas adecuadas para su análisis. OpenSearch UI también trabaja con Amazon Q Developer para proporcionar el chat de Amazon Q y ayudar a generar visualizaciones, resúmenes de alertas, información y detectores de anomalías recomendados.
- Compatibilidad con varios lenguajes de consulta: OpenSearch UI es compatible con el lenguaje de procesamiento de canalizaciones (PPL), SQL, Lucene y el lenguaje de consultas de paneles (DQL).

- Soporte multiregional y multicuenta: OpenSearch UI puede utilizar la característica de búsqueda entre clústeres para conectarse con dominios de OpenSearch de diferentes cuentas y regiones a fin de realizar análisis y visualizaciones agregados.

Para empezar y crear su primera interfaz de usuario de OpenSearch UI, siga las instrucciones que se indican en the section called “Introducción”.

Para obtener información acerca de las últimas características publicadas para OpenSearch UI, consulte the section called “Historial de versiones”.

Temas

- Historial de versiones de la interfaz de usuario de Amazon OpenSearch Service
- Introducción a la interfaz de usuario de OpenSearch en Amazon OpenSearch Service
- Habilitar la federación SAML con AWS Identity and Access Management
- Administración de las asociaciones de orígenes de datos y los permisos de acceso a la nube privada virtual
- Uso de los espacios de trabajo de Amazon OpenSearch Service
- Acceso a datos entre regiones y entre cuentas con búsqueda entre clústeres
- Administración del acceso a OpenSearch UI desde un punto de conexión de VPC
- Puntos de conexión y cuotas de OpenSearch UI

Historial de versiones de la interfaz de usuario de Amazon OpenSearch Service

En la siguiente tabla, se muestran todas las versiones del soporte de Amazon OpenSearch Service para OpenSearch UI y las características y mejoras incluidas en cada versión.

Cambio	Fecha de lanzamiento de la nueva versión	Descripción
Nueva característica	2025-04-16	OpenSearch UI ahora funciona con la búsqueda entre clústeres. Esto le permite utilizar OpenSearch UI en una sola Región de AWS para acceder a los clústeres de una

Cambio	Fecha de lanzamiento de la nueva versión	Descripción
		región diferente. Esto se hace configurándolo como un clúster remoto conectado a un clúster de la misma región. Para obtener más información, consulte the section called “Acceso a datos entre regiones y entre cuentas con búsqueda entre clústeres”.
Nueva característica	2025-03-31	Amazon Q Developer sin servidor ya está disponible con carácter general en Amazon OpenSearch Service. Para obtener más información, consulte Compatibilidad con Amazon Q.
Nueva característica	2025-02-05	La federación del lenguaje de aserción de seguridad 2.0 (SAML) mediante AWS Identity and Access Management (IAM) ahora funciona con OpenSearch UI. Esto permite crear una experiencia de inicio de sesión único (SSO) iniciada por un proveedor de identidad para los usuarios finales. Para obtener más información, consulte the section called “Habilitar la federación SAML con IAM”.
Nueva integración	2024-12-01	La integración sin ETL con Amazon CloudWatch simplifica el análisis y la visualización de los datos de registro, lo que reduce los gastos técnicos y operativos. Para obtener más información, consulte El nuevo Amazon CloudWatch y Amazon OpenSearch Service lanzan una experiencia de análisis integrada en el blog de novedades de AWS.
Nueva integración	2024-12-01	La integración sin ETL con Amazon Security Lake permite a las organizaciones buscar, analizar y obtener información procesable a partir de sus datos de seguridad de manera eficiente. Para obtener más información, consulte Introducción a la integración de Amazon OpenSearch Service y Amazon Security Lake para simplificar los análisis de seguridad en el blog de novedades de AWS.

Cambio	Fecha de lanzamiento de la nueva versión	Descripción
Versión inicial	2024-11-07	La versión pública inicial de OpenSearch UI. Para obtener más información, consulte Amazon OpenSearch Service lanza la interfaz de usuario OpenSearch de próxima generación en el blog de AWS Big Data.

Introducción a la interfaz de usuario de OpenSearch en Amazon OpenSearch Service

En Amazon OpenSearch Service, una aplicación es una instancia de la interfaz de usuario de OpenSearch (OpenSearch UI). Cada aplicación se puede asociar a varios orígenes de datos y un solo origen se puede asociar a varias aplicaciones. Puede crear varias aplicaciones para distintos administradores mediante diferentes opciones de autenticación compatibles.

Utilice la información de este tema como guía durante el proceso de creación de una aplicación de OpenSearch UI mediante la Consola de administración de AWS o la AWS CLI.

Temas

- Permisos necesarios para crear aplicaciones de Amazon OpenSearch Service
- Creación de una aplicación de OpenSearch UI
- Administrar los administradores de aplicaciones

Permisos necesarios para crear aplicaciones de Amazon OpenSearch Service

Antes de crear una aplicación, compruebe que se le han concedido los permisos necesarios para la tarea. Si es necesario, póngase en contacto con un administrador de cuentas para obtener ayuda.

Permisos generales

Para trabajar con aplicaciones en OpenSearch Service, necesita los permisos que se muestran en la siguiente política. Los permisos sirven para los siguientes propósitos:

- Los cinco permisos `es:*Application` son necesarios para crear y administrar una aplicación.
- Los tres permisos `es:*Tags` son necesarios para agregar, enumerar y eliminar etiquetas de la aplicación.
- Los permisos `aoss:BatchGetCollection`, `es:DescribeDomain` y `es:GetDirectQueryDataSource` son necesarios para asociar los orígenes de datos.
- Los permisos `aoss:APIAccessAll`, `es:ESHttp*` y 4 `opensearch:*DirectQuery*` son necesarios para acceder a los orígenes de datos.
- `iam:CreateServiceLinkedRole` proporciona permiso a Amazon OpenSearch Service para crear un rol vinculado al servicio (SLR) en su cuenta. Esta función se utiliza y permite que la aplicación de interfaz de usuario OpenSearch publique las métricas de Amazon CloudWatch en su cuenta. Para obtener más información, consulte the section called “Permisos” en el tema Uso de roles vinculados al servicio para crear dominios de VPC y orígenes de datos de consulta directa.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "es:CreateApplication",
                "es:DeleteApplication",
                "es:GetApplication",
                "es:ListApplications",
                "es:UpdateApplication",
                "es:AddTags",
                "es:ListTags",
                "es:RemoveTags",
                "aoss:APIAccessAll",
                "es:ESHttp*",
                "opensearch:StartDirectQuery",
                "opensearch:GetDirectQuery",
                "opensearch:CancelDirectQuery",
                "opensearch:GetDirectQueryResult",
                "aoss:BatchGetCollection",
                "aoss:ListCollections",
                "es:DescribeDomain",
```

```
                "es:DescribeDomains",
                "es:ListDomainNames",
                "es:GetDirectQueryDataSource",
                "es:ListDirectQueryDataSources"
            ],
            "Resource": "*"
        },
        {
            "Sid": "VisualEditor1",
            "Effect": "Allow",
            "Action": "iam:CreateServiceLinkedRole",
            "Resource": "arn:aws:iam::*:role/aws-service-role/
opensearchservice.amazonaws.com/AWSServiceRoleForAmazonOpenSearchService"
        }
    ]
}
```

Permisos para crear una aplicación que utilice la autenticación del Centro de identidades de IAM (opcional)

De forma predeterminada, las aplicaciones del panel de control se autentican mediante AWS Identity and Access Management (IAM) para administrar los permisos de los usuarios de los recursos AWS. Sin embargo, puede optar por ofrecer una experiencia de inicio de sesión único mediante el Centro de identidades de IAM, que le permite utilizar sus proveedores de identidad actuales para iniciar sesión en las aplicaciones de OpenSearch UI. En este caso, seleccionará la opción Autenticación con el Centro de identidades de IAM en el procedimiento que aparece más adelante en este tema y, a continuación, concederá a los usuarios del Centro de identidades de IAM los permisos necesarios para acceder a la aplicación de OpenSearch UI.

Para crear una aplicación que utilice la autenticación del Centro de identidades de IAM, necesitará los siguientes permisos. Reemplace los *valores de marcador de posición* con su información. Si es necesario, póngase en contacto con un administrador de cuentas para obtener ayuda.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
```

```
        {
            "Sid": "IDCPermissions",
            "Effect": "Allow",
            "Action": [
                "es:CreateApplication",
                "es:DeleteApplication",
                "es:GetApplication",
                "es:ListApplications",
                "es:UpdateApplication",
                "es:AddTags",
                "es:ListTags",
                "es:RemoveTags",
                "aoss:BatchGetCollection",
                "aoss:ListCollections",
                "es:DescribeDomain",
                "es:DescribeDomains",
                "es:ListDomainNames",
                "es:GetDirectQueryDataSource",
                "es:ListDirectQueryDataSources",
                "sso:CreateApplication",
                "sso:DeleteApplication",
                "sso:PutApplicationGrant",
                "sso:PutApplicationAccessScope",
                "sso:PutApplicationAuthenticationMethod",
                "sso:ListInstances",
                "sso:DescribeApplicationAssignment",
                "sso:DescribeApplication",
                "sso:CreateApplicationAssignment",
                "sso:ListApplicationAssignments",
                "sso:DeleteApplicationAssignment",
                "sso-directory:SearchGroups",
                "sso-directory:SearchUsers",
                "sso:ListDirectoryAssociations",
                "identitystore:DescribeUser",
                "identitystore:DescribeGroup",
                "iam:ListRoles"
            ],
            "Resource": "*"
        },
        {
            "Sid": "SLRPermission",
            "Effect": "Allow",
            "Action": "iam:CreateServiceLinkedRole",
```

```
            "Resource": "arn:aws:iam::*:role/aws-service-role/
opensearchservice.amazonaws.com/AWSServiceRoleForAmazonOpenSearchService"
        },
        {
            "Sid": "PassRolePermission",
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::111122223333:role/iam-role-for-identity-
center"
        }
    ]
}
```

Creación de una aplicación de OpenSearch UI

Cree una aplicación que especifique el nombre de la aplicación, el método de autenticación y los administradores mediante uno de los siguientes procedimientos.

Temas

- Creación de una aplicación de OpenSearch UI que utilice la autenticación de IAM en la consola
- Creación de una aplicación de OpenSearch UI que utilice la autenticación de AWS IAM Identity Center en la consola
- Creación de una aplicación de OpenSearch UI que utilice la autenticación de AWS IAM Identity Center mediante la AWS CLI

Creación de una aplicación de OpenSearch UI que utilice la autenticación de IAM en la consola

Para crear una aplicación de OpenSearch UI que utilice la autenticación de IAM en la consola

1. Inicie sesión en la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. En el panel de navegación de la izquierda, seleccione OpenSearch UI (Dashboards).
3. Elija Creación de aplicación.
4. En Nombre de la aplicación, especifique un nombre para su aplicación.
5. No active la casilla de verificación Autenticación con el Centro de identidades de IAM. Para obtener información sobre cómo crear una aplicación con autenticación mediante AWS IAM

Identity Center, consulte the section called “Creación de una aplicación de OpenSearch UI que utilice la autenticación de AWS IAM Identity Center en la consola” más adelante en este tema.

6. (Opcional) Se lo agrega automáticamente como administrador de la aplicación que está creando. En el área de administración de admin de la aplicación OpenSearch, puede conceder permisos de administrador a otros usuarios.

> **Note**
>
> El rol de administrador de la aplicación de OpenSearch UI otorga permisos para editar y eliminar una aplicación de OpenSearch UI. Los administradores de aplicaciones también pueden crear, editar y eliminar espacios de trabajo en una aplicación de OpenSearch UI.

   Para conceder permisos de administrador a otros usuarios, elija una de las siguientes opciones:

   - Otorgue el permiso de administrador a usuarios específicos: en el campo de admin de la aplicación OpenSearch, en la lista emergente Propiedades, seleccione usuarios de IAM o

     usuarios de AWS IAM Identity Center y, a continuación, elija los usuarios individuales a los que va a conceder los permisos de administrador.
   - Otorgue permisos de administrador a todos los usuarios: todos los usuarios de su organización o cuenta tienen permisos de administrador.

7. (Opcional) En el área Etiquetas, aplique a la pila uno o varios pares de nombre-valor de claves de etiqueta.

   Las etiquetas son metadatos opcionales que usted asigna a un recurso. Las etiquetas permiten clasificar los recursos de diversas maneras, por ejemplo, según la finalidad, el propietario o el entorno.

8. Elija Crear.

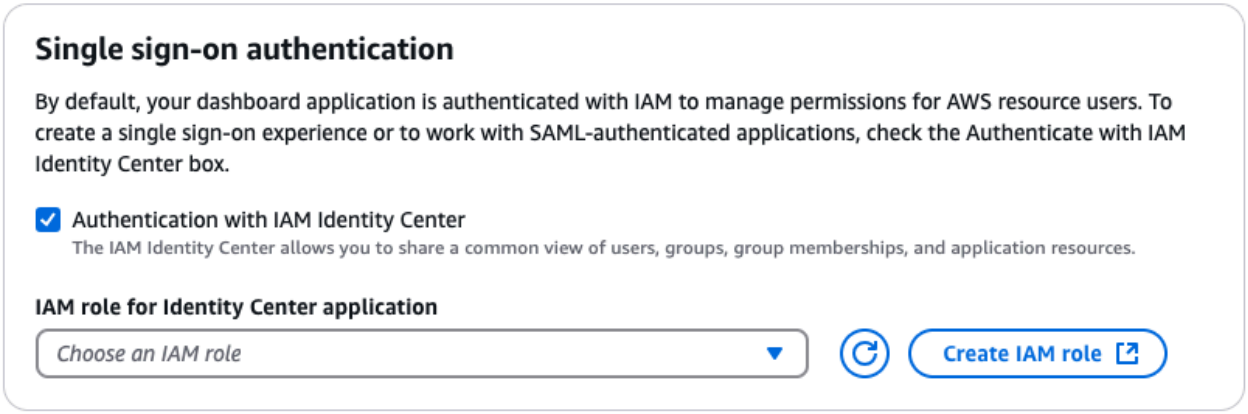
Creación de una aplicación de OpenSearch UI que utilice la autenticación de AWS IAM Identity Center en la consola

Para crear una aplicación de OpenSearch UI que utilice la autenticación de AWS IAM Identity Center, debe tener los permisos de IAM descritos anteriormente en este tema en the section called “Permisos para crear una aplicación que utilice la autenticación del Centro de identidades de IAM (opcional)”.

Para crear una aplicación de OpenSearch UI que utilice la autenticación de AWS IAM Identity Center en la consola

1. Inicie sesión en la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. En el panel de navegación de la izquierda, seleccione OpenSearch UI (Dashboards).
3. Elija Creación de aplicación.
4. En Nombre de la aplicación, especifique un nombre para su aplicación.
5. (Opcional) Para habilitar el inicio de sesión único en su organización o cuenta, haga lo siguiente:

   a. Seleccione la casilla Autenticación con el Centro de identidades de IAM, como se muestra en la siguiente imagen:



   b. Realice una de las siguientes acciones:

   - En la lista Rol de IAM para la aplicación del Centro de identidades de IAM, elija un rol de IAM existente que proporcione los permisos necesarios para que el Centro de identidades de IAM acceda a OpenSearch UI y a los orígenes de datos asociados. Consulte las políticas en el siguiente punto para ver los permisos que debe tener el rol.
   - Creación de un rol de IAM con los permisos necesarios. Utilice los siguientes procedimientos de la Guía del usuario de IAM con las opciones especificadas para crear un nuevo rol y con la política de permisos y la política de confianza necesarias.
     - Procedimiento: Creación de políticas de IAM (consola)

       Siguiendo los pasos de este procedimiento, pegue la siguiente política en el campo JSON del editor de políticas:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "IdentityStoreOpenSearchDomainConnectivity",
            "Effect": "Allow",
            "Action": [
                "identitystore:DescribeUser",
                "identitystore:ListGroupMembershipsForMember",
                "identitystore:DescribeGroup"
            ],
            "Resource": "*",
            "Condition": {
                "ForAnyValue:StringEquals": {
                    "aws:CalledViaLast": "es.amazonaws.com"
                }
            }
        },
        {
            "Sid": "OpenSearchDomain",
            "Effect": "Allow",
            "Action": [
                "es:ESHttp*"
            ],
            "Resource": "*"
        },
        {
            "Sid": "OpenSearchServerless",
            "Effect": "Allow",
            "Action": [
                "aoss:APIAccessAll"
            ],
            "Resource": "*"
        }
    ]
}
```

- Procedimiento: Cree un rol mediante políticas de confianza personalizadas

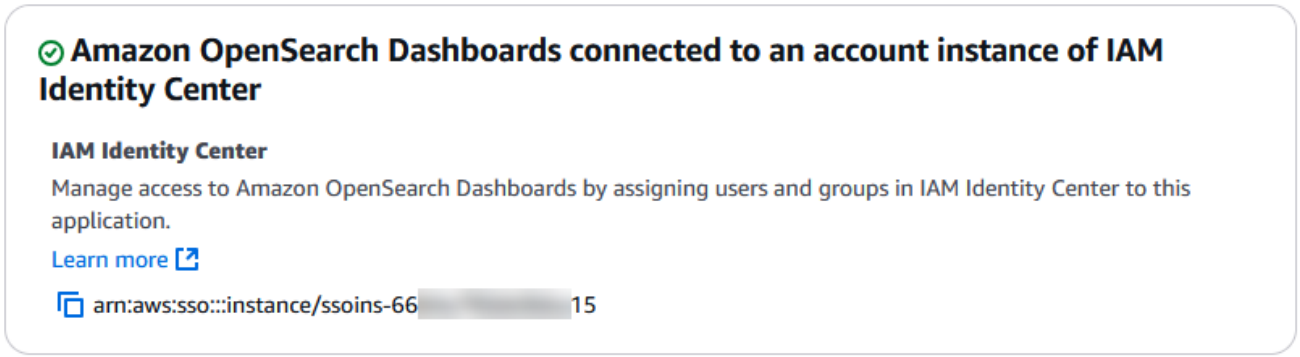
  Siguiendo los pasos de este procedimiento, reemplace el marcador de posición JSON del cuadro Política de confianza personalizada con lo siguiente:

> **Tip**
>
> Si va a agregar la política de confianza a un rol existente, agregue la política en la pestaña Relación de confianza del rol.

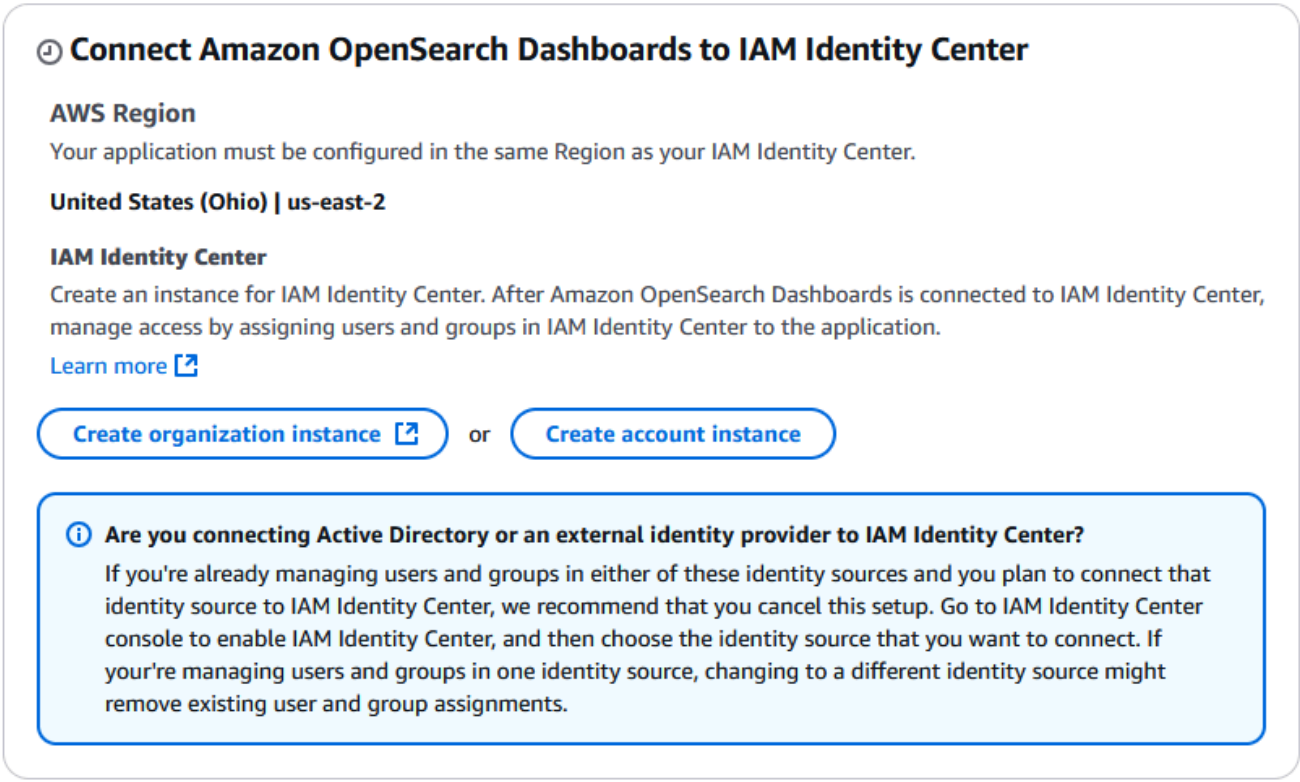
JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service":
 "application.opensearchservice.amazonaws.com"
            },
            "Action": "sts:AssumeRole"
        },
        {
            "Effect": "Allow",
            "Principal": {
                "Service":
 "application.opensearchservice.amazonaws.com"
            },
            "Action": "sts:SetContext",
            "Condition": {
                "ForAllValues:ArnEquals": {
                "sts:RequestContextProviders":
 "arn:aws:iam::123456789012:oidc-provider/portal.sso.us-
east-1.amazonaws.com/apl/application-id"
                }
            }
        }
    ]
}
```

c. Si ya se ha creado una instancia del Centro de identidades de IAM en su organización o cuenta, la consola informa de que Amazon OpenSearch Dashboards ya está conectado a una instancia de organización del Centro de identidades de IAM, como se muestra en la siguiente imagen.



Si el Centro de Identidad de IAM aún no está disponible en su organización o cuenta, usted o un administrador con los permisos necesarios pueden crear una instancia de organización o de cuenta. El área Conectar Amazon OpenSearch Dashboards al Centro de identidades de IAM ofrece opciones para ambos, como se muestra en la siguiente imagen:

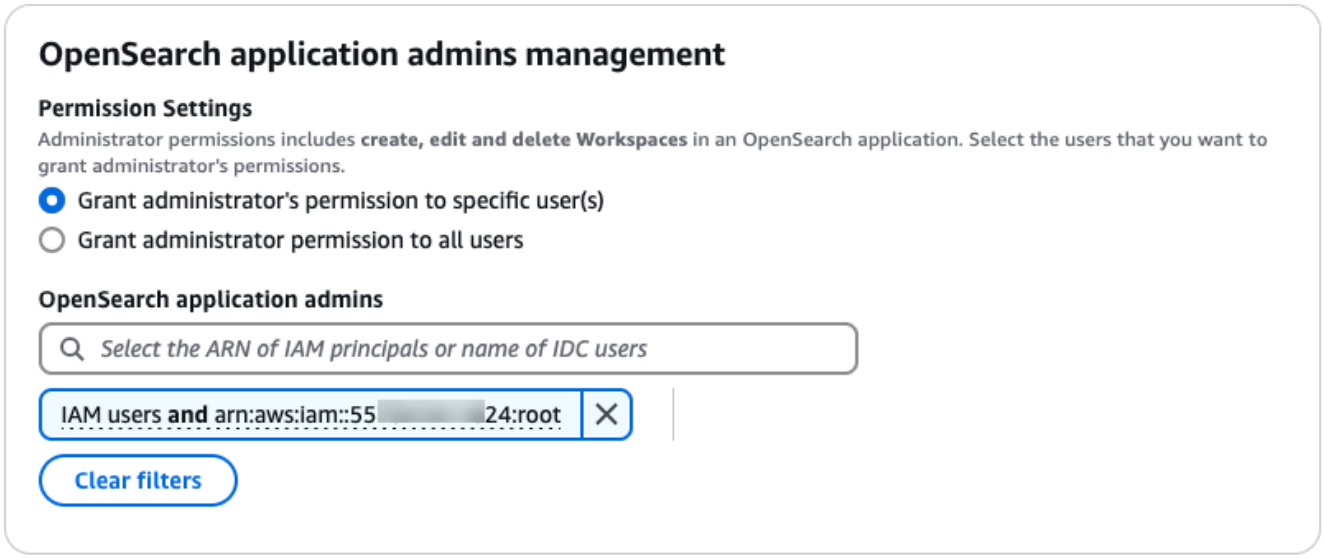


En este caso, puede crear una instancia de cuenta en el Centro de Identidad de IAM para realizar pruebas o solicitar que un administrador cree una instancia de organización en el Centro de Identidad de IAM. Para obtener más información, consulte los siguientes temas en la Guía del usuario de AWS IAM Identity Center:

> **Note**
>
> Por el momento, las aplicaciones de OpenSearch UI solo se pueden crear en la misma Región de AWS que la instancia de Centro de identidades de IAM de su organización. Para obtener información sobre cómo acceder a los orígenes de datos en esa región después de crear la aplicación, consulte the section called “Acceso a datos entre regiones y entre cuentas con búsqueda entre clústeres”.

- Instancias de organización de IAM Identity Center
- Instancias de cuenta de IAM Identity Center
- Habilitar AWS IAM Identity Center

6. (Opcional) Se lo agrega automáticamente como administrador de la aplicación que está creando. En el área de administración de admin de la aplicación OpenSearch, puede conceder permisos de administrador a otros usuarios, como se muestra en la siguiente imagen:



> **Note**
>
> El rol de administrador de la aplicación de OpenSearch UI otorga permisos para editar y eliminar una aplicación de OpenSearch UI. Los administradores de aplicaciones también pueden crear, editar y eliminar espacios de trabajo en una aplicación de OpenSearch UI.

Para conceder permisos de administrador a otros usuarios, elija una de las siguientes opciones:

- Otorgue el permiso de administrador a usuarios específicos: en el campo de admin de la aplicación OpenSearch, en la lista emergente Propiedades, seleccione usuarios de IAM o usuarios de AWS IAM Identity Center y, a continuación, elija los usuarios individuales a los que va a conceder los permisos de administrador.
- Otorgue permisos de administrador a todos los usuarios: todos los usuarios de su organización o cuenta tienen permisos de administrador.

7. (Opcional) En el área Etiquetas, aplique a la pila uno o varios pares de nombre-valor de claves de etiqueta.

   Las etiquetas son metadatos opcionales que usted asigna a un recurso. Las etiquetas permiten clasificar los recursos de diversas maneras, por ejemplo, según la finalidad, el propietario o el entorno.

8. Elija Crear.

Creación de una aplicación de OpenSearch UI que utilice la autenticación de AWS IAM Identity Center mediante la AWS CLI

Para crear una aplicación de OpenSearch UI que utilice la autenticación AWS IAM Identity Center mediante la AWS CLI, utilice el comando create-application con las siguientes opciones:

- `--name`: nombre de la aplicación.
- `--iam-identity-center-options`: (opcional) la instancia del Centro de identidades de IAM y el rol de IAM que OpenSearch utilizará para la autenticación y el control de acceso.

Reemplace los *valores de marcador de posición* con su información.

```
aws opensearch create-application \
    --name application-name \
    --iam-identity-center-options "
          {
          \"enabled\":true,
          \"iamIdentityCenterInstanceArn\":\"arn:aws:sso:::instance/sso-instance\",
          \"iamRoleForIdentityCenterApplicationArn\":\"arn:aws:iam::account-
id:role/role-name\"
```

```
        }
    "
```

Administrar los administradores de aplicaciones

Un administrador de aplicaciones de OpenSearch UI es un rol definido con permiso para editar y eliminar una aplicación de OpenSearch UI.

De forma predeterminada, como creador de una aplicación de OpenSearch UI, usted es el primer administrador de la aplicación de OpenSearch UI.

Administración de los administradores de OpenSearch UI mediante la consola

Puede agregar administradores adicionales a una aplicación de OpenSearch UI en la Consola de administración de AWS, ya sea durante el flujo de trabajo de creación de la aplicación o en la página Editar una vez creada la aplicación.

El rol de administrador de la aplicación de OpenSearch UI otorga permisos para editar y eliminar una aplicación de OpenSearch UI. Los administradores de aplicaciones también pueden crear, editar y eliminar espacios de trabajo en una aplicación de OpenSearch UI.

En la página de detalles de la aplicación, puede buscar el nombre de recurso de Amazon (ARN) de una entidad principal de IAM o el nombre de un usuario del Centro de Identidad de IAM.

Para administrar a los administradores de OpenSearch UI mediante la consola

1. Inicie sesión en la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. En el panel de navegación de la izquierda, seleccione OpenSearch UI (Dashboards).
3. En el área de aplicaciones de OpenSearch, elija el nombre de una aplicación existente.
4. Elija Editar
5. Para conceder permisos de administrador a otros usuarios, elija una de las siguientes opciones:

   - Otorgue el permiso de administrador a usuarios específicos: en el campo de admin de la aplicación OpenSearch, en la lista emergente Propiedades, seleccione usuarios de IAM o

     usuarios de AWS IAM Identity Center y, a continuación, elija los usuarios individuales a los que va a conceder los permisos de administrador.

- Otorgue permisos de administrador a todos los usuarios: todos los usuarios de su organización o cuenta tienen permisos de administrador.

6. Elija Actualizar.

Puede eliminar administradores adicionales, pero cada aplicación de OpenSearch UI debe tener al menos un administrador.

Administración de los administradores de OpenSearch UI mediante la AWS CLI

Puede crear y actualizar los administradores de aplicaciones de OpenSearch UI mediante la AWS CLI.

Creación de administradores de OpenSearch UI mediante la AWS CLI

Los siguientes son ejemplos de cómo agregar entidades principales de IAM y usuarios de Centro de identidades de IAM como administradores al crear una aplicación de OpenSearch UI.

Ejemplo 1: Crear una aplicación de OpenSearch UI que agregue un usuario de IAM como administrador

Ejecute el siguiente comando para crear una aplicación de OpenSearch Search que agregue un usuario de IAM como administrador. Reemplace los *valores de marcador de posición* con su información.

```
aws opensearch create-application \
    --name application-name \
    --app-configs "
        {
        \"key\":\"opensearchDashboards.dashboardAdmin.users\",
        \"value\":\"arn:aws:iam::account-id:user/user-id\"
        }
    "
```

Ejemplo 2: Crear una aplicación de OpenSearch UI que habilite Centro de identidades de IAM y agregue un ID de usuario de Centro de identidades de IAM como administrador de la aplicación de OpenSearch UI

Ejecute el siguiente comando para crear una aplicación de OpenSearch UI que habilite Centro de identidades de IAM y agregue un ID de usuario de Centro de identidades de IAM como administrador

de la aplicación de OpenSearch UI. Reemplace los *valores de marcador de posición* con su información.

`key` especifica el elemento de configuración que se va a establecer, como el rol de administrador de la aplicación de OpenSearch UI. Los valores válidos son `opensearchDashboards.dashboardAdmin.users` y `opensearchDashboards.dashboardAdmin.groups`.

xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx representa el valor asignado a la clave, como el nombre de recurso de Amazon (ARN) de un usuario de IAM.

```
aws opensearch create-application \
    --name myapplication \
    --iam-identity-center-options "
        {
        \"enabled\":true,
        \"iamIdentityCenterInstanceArn\":\"arn:aws:sso:::instance/ssoins-instance-id\",
        \"iamRoleForIdentityCenterApplicationArn\":\"arn:aws:iam::account-id:role/role-
name\"
        }
    " \
    --app-configs "
        {
        \"key\":\"opensearchDashboards.dashboardAdmin.users\",
        \"value\":\"xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx\"
        }
    "
```

Actualización de administradores de OpenSearch UI mediante la AWS CLI

A continuación se muestran ejemplos de cómo actualizar las entidades principales de IAM y los usuarios del Centro de Identidad de IAM asignados como administradores de una aplicación OpenSearch existente.

Ejemplo 1: Agregar un usuario de IAM como administrador de una aplicación de OpenSearch existente

Ejecute el siguiente comando para actualizar una aplicación de OpenSearch UI y agregar un usuario de IAM como administrador. Reemplace los *valores de marcador de posición* con su información.

```
aws opensearch update-application \
```

```
    --id myapplication \
    --app-configs "
        {
        \"key\":\"opensearchDashboards.dashboardAdmin.users\",
        \"value\":\"arn:aws:iam::account-id:user/user-id\"
        }
    "
```

Ejemplo 2: Actualizar una aplicación de OpenSearch UI para agregar un ID de usuario de Centro de identidades de IAM como administrador de la aplicación de OpenSearch UI

Ejecute el siguiente comando para actualizar una aplicación de OpenSearch UI y agregar un ID de usuario de Centro de identidades de IAM como administrador de la aplicación de OpenSearch UI. Reemplace los *valores de marcador de posición* con su información.

`key` especifica el elemento de configuración que se va a establecer, como el rol de administrador de la aplicación de OpenSearch UI. Los valores válidos son `opensearchDashboards.dashboardAdmin.users` y `opensearchDashboards.dashboardAdmin.groups`.

xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx representa el valor asignado a la clave, como el nombre de recurso de Amazon (ARN) de un usuario de IAM.

```
aws opensearch update-application \
    --id myapplication \
    --app-configs "
        {
        \"key\":\"opensearchDashboards.dashboardAdmin.users\",
        \"value\":\"xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx\"
        }
    "
```

Habilitar la federación SAML con AWS Identity and Access Management

OpenSearch UI admite la federación de identidades con SAML 2.0, un estándar abierto que utilizan muchos proveedores de identidad (IdP). Esto permite la federación de identidades con (IAM) AWS Identity and Access Management. Con este soporte, los usuarios de su cuenta u organización pueden acceder directamente a OpenSearch UI asumiendo roles de IAM. Puede crear

una experiencia de inicio de sesión único iniciada por el proveedor de identidad (IdP) para sus usuarios finales, en la que puedan autenticarse en el proveedor de identidad externo y ser redirigidos directamente a la página definida en OpenSearch UI. También puede implementar un control de acceso detallado si configura sus usuarios finales o grupos para que asuman diferentes roles de IAM con distintos permisos de acceso a OpenSearch UI y a los orígenes de datos asociados.

En este tema se muestran instrucciones paso a paso para configurar el uso de SAML con OpenSearch UI. En estos procedimientos, utilizamos como ejemplo los pasos para configurar la aplicación de administración de identidades y accesos de Okta. Los pasos de configuración para otros proveedores de identidad, como Azure Active Directory y Ping, son similares.

Temas

- Paso 1: Configurar la aplicación de proveedor de identidades (Okta)
- Paso 2: Configurar AWS para Okta
- Paso 3: Creación de la política de acceso de Amazon OpenSearch Service en IAM
- Paso 4: Compruebe la experiencia de inicio de sesión único iniciada por el proveedor de identidad con SAML
- Paso 5: Configuración del control de acceso detallado basado en atributos de SAML

Paso 1: Configurar la aplicación de proveedor de identidades (Okta)

Para usar SAML con OpenSearch UI, el primer paso es configurar el proveedor de identidad.

Tarea 1: creación de usuarios de Okta

1. Inicie sesión en su organización de Okta en https://login.okta.com/ como usuario con privilegios administrativos.
2. En la consola de administración, en Directorio, en el panel de navegación, selecciona Personas.
3. Elija Agregar persona.
4. En Nombre, introduzca el nombre del usuario.
5. En Apellido, introduzca el apellido del usuario.
6. En Usuario, introduzca el nombre de usuario del usuario en formato de correo electrónico.
7. Elija Estableceré la contraseña e introduzca una contraseña
8. (Opcional) Desactive la casilla El usuario debe cambiar la contraseña la primera vez que inicie sesión si no desea que el usuario cambie la contraseña la primera vez que inicie sesión.

9. Seleccione Save.

Tarea 2: crear y asignar grupos

1. Inicie sesión en su organización de Okta en https://login.okta.com/ como usuario con privilegios administrativos.
2. En el panel de navegación de Okta, seleccione Directorio y, a continuación, Grupos.
3. Elija Agregar grupo.
4. Escriba un nombre y elija Crear grupo de registros.
5. Elija el grupo recién creado y, a continuación, seleccione Asignar personas.
6. Seleccione el signo más (+) y, a continuación, seleccione Listo.
7. (Opcional) Repita los pasos 1 a 6 para agregar más grupos.

Tarea 3: Crear aplicaciones de Okta

1. Inicie sesión en su organización de Okta en https://login.okta.com/ como usuario con privilegios administrativos.
2. En la consola de administración, en Aplicaciones, en el panel de navegación, seleccione Aplicaciones.
3. Elija Crear integración de aplicaciones.
4. Elija SAML 2.0 como método de inicio de sesión y, a continuación, seleccione Siguiente.
5. Escriba un nombre para su grupo (por ejemplo, **`OpenSearch_UI`**) y elija Siguiente.
6. Introduzca los siguientes valores en la aplicación; no es necesario que cambie otros valores:
    a. 1. En el caso de la URL de inicio de sesión única, introduzca **`https://signin.aws.amazon.com/saml`** para las regiones comerciales de AWS o la URL específica de su región.
    b. 2. Para URI de audiencia (ID de entidad del SP), introduzca **`urn:amazon:webservices`**.
    c. 3. En Formato de nombre, ingrese **`EmailAddress`**.
7. Elija Siguiente.
8. Seleccione Soy cliente de Okta y agrego una aplicación interna y, a continuación, seleccione Esta es una aplicación interna que hemos creado.
9. Seleccione Finalizar.

10. Elija Asignar y, luego, elija Asignar a personas.
11. Seleccione Asignar a grupos y, a continuación, seleccione Asignar junto a los grupos que desee agregar.
12. Seleccione Listo.

Tarea 4: Configurar la configuración avanzada de Okta

Después de crear la aplicación SAML personalizada, complete los pasos siguientes:

1. Inicie sesión en su organización de Okta en https://login.okta.com/ como usuario con privilegios administrativos.

   En la consola de administración, en el área General, seleccione Editar en la configuración de SAML.
2. Elija Siguiente.
3. Defina Estado de retransmisión predeterminado en el punto de conexión de OpenSearch UI con el siguiente formato:

   `https://region.console.aws.amazon.com/aos/home?region=region#opensearch/applications/application-id/redirectToDashboardURL`.

   A continuación, se muestra un ejemplo:

   `https://us-east-2.console.aws.amazon.com/aos/home?region=us-east-2#opensearch/applications/abc123def4567EXAMPLE/redirectToDashboardURL`
4. En Instrucciones de atributo de grupo (opcional), agregue el siguiente atributo:

   a. Proporcione el rol de IAM y el proveedor de identidad en un formato separado por comas mediante el atributo Role. Utilizará este mismo rol de IAM y proveedor de identidad en un paso posterior al configurar la configuración AWS.

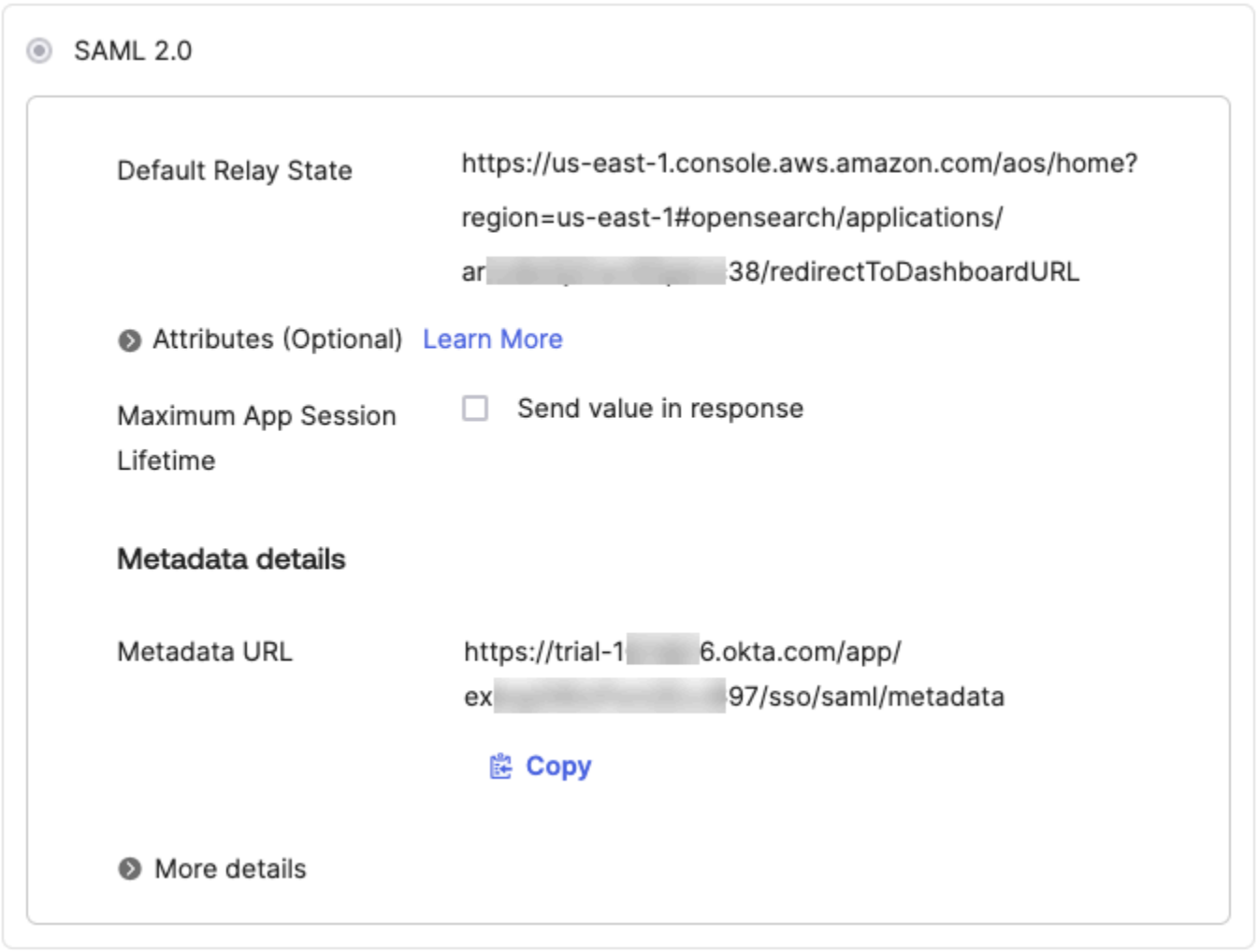
   b. Defina user.login como RoleSessionName. Se utiliza como identificador para las credenciales temporales que se emiten cuando se asume el rol.

   De referencia:

Nombre	Formatos de nombre	Formato	Ejemplo
`https://aws.amazon.com/SAML/Attributes/Role`	Sin especificar	`arn:aws:iam:: aws-account-id :role/role-name,arn:aws:iam:: aws-account-id :saml-provider/ provider-name`	`arn:aws:iam::111222333444:role/oktarole,arn:aws:iam::111222333444:saml-provider/oktaidp`
`https://aws.amazon.com/SAML/Attributes/RoleSessionName`	Sin especificar	`user.login`	`user.login`

5. Tras agregar las propiedades del atributo, elija Siguiente y, a continuación, elija Finalizar.

Los atributos deben tener un formato similar al que se muestra en la siguiente imagen. El valor de estado de retransmisión predeterminado es la URL que define la página de destino para los usuarios finales de su cuenta u organización una vez que hayan completado la validación del inicio de sesión único desde Okta. Puede configurarlo en cualquier página de OpenSearch UI y, a continuación, proporcionar esa URL a los usuarios finales previstos.



Paso 2: Configurar AWS para Okta

Complete las siguientes tareas para la configuración de AWS para Okta.

Tarea 1: recopile información de Okta

Para este paso, tendrá que recopilar su información de Okta para poder configurarla más adelante en AWS.

1. Inicie sesión en su organización de Okta en https://login.okta.com/ como usuario con privilegios administrativos.
2. En la pestaña Iniciar sesión, en la esquina inferior derecha de la página, seleccione Ver las instrucciones de configuración de SAML.

3. Copie o almacene el valor para URL de inicio de sesión único del proveedor de identidades. Puede usar esta URL cuando se conecte a cualquier cliente SQL de terceros, como SQL Workbench/J.
4. Utilice los metadatos del proveedor de identidad del bloque 4 y, a continuación, guarde el archivo de metadatos en formato.xml (por ejemplo, `metadata.xml`).

Tarea 2: Crear el proveedor de IAM

Para crear su proveedor de IAM, complete los siguientes pasos.

1. Inicie sesión en Consola de administración de AWS y abra la consola IAM en https://console.aws.amazon.com/iam/.
2. En el panel de navegación, elija Proveedores de identidad en Administración de acceso.
3. Elija Agregar proveedor.
4. Para Tipo de proveedor: seleccione SAML.
5. En Nombre del proveedor, escriba un nombre.
6. Para el documento de metadatos, elija Elegir archivo y cargue el archivo de metadatos (.xml) que descargó anteriormente.
7. Elija Agregar proveedor.

Tarea 3: Creación de un rol de IAM

Para crear su primer rol AWS Identity and Access Management, siga los pasos que se describen a continuación.

1. Inicie sesión en Consola de administración de AWS y abra la consola IAM en https://console.aws.amazon.com/iam/.
2. En el panel de navegación, en Administración del acceso, elija Roles.
3. Seleccione Crear rol.
4. En Tipo de entidad de confianza, seleccione Federación SAML 2.0.
5. En el proveedor basado en SAML 2.0, elija el proveedor de identidades que ha creado en IAM.
6. Seleccione Permitir acceso mediante programación y mediante la Consola de administración de AWS.
7. Elija Siguiente.

8. En la lista de políticas de permisos, active las casillas de verificación de la política que creó anteriormente y de OpenSearchFullAccess.
9. Elija Siguiente.
10. En la sección Revisar, en Nombre de rol, escriba un nombre para el rol (por ejemplo, **`oktarole`**).
11. (Opcional) En Descripción de imagen, ingrese una breve descripción del propósito del rol.
12. Seleccione Crear rol.
13. Vaya al rol que acaba de crear, seleccione la pestaña Relaciones de confianza y, a continuación, elija Editar política de confianza.
14. En el panel Editar declaración, en Agregar acciones para STS, active la casilla TagSession.
15. Elija Actualizar política.

Paso 3: Creación de la política de acceso de Amazon OpenSearch Service en IAM

Aprenda a configurar sus funciones de IAM para el control de acceso de OpenSearch. Con los roles de IAM, puede implementar un control de acceso detallado para que sus grupos de usuarios de Okta accedan a los recursos de OpenSearch. En este tema, se muestra la configuración basada en roles de IAM mediante dos grupos de ejemplo.

Sample group: Alice

Solicitud:

```
GET _plugins/_security/api/roles/alice-group
```

Resultado:

```
{
  "alice-group": {
    "reserved": false,
    "hidden": false,
    "cluster_permissions": [
      "unlimited"
    ],
    "index_permissions": [
      {
```

```
        "index_patterns": [
          "alice*"
        ],
        "dls": "",
        "fls": [],
        "masked_fields": [],
        "allowed_actions": [
          "indices_all"
        ]
      }
    ],
    "tenant_permissions": [
      {
        "tenant_patterns": [
          "global_tenant"
        ],
        "allowed_actions": [
          "kibana_all_write"
        ]
      }
    ],
    "static": false
  }
}
```

Sample group: Bob

Solicitud:

```
GET _plugins/_security/api/roles/bob-group
```

Resultado:

```
{
  "bob-group": {
    "reserved": false,
    "hidden": false,
    "cluster_permissions": [
      "unlimited"
    ],
    "index_permissions": [
      {
        "index_patterns": [
```

```
          "bob*"
        ],
        "dls": "",
        "fls": [],
        "masked_fields": [],
        "allowed_actions": [
          "indices_all"
        ]
      }
    ],
    "tenant_permissions": [
      {
        "tenant_patterns": [
          "global_tenant"
        ],
        "allowed_actions": [
          "kibana_all_write"
        ]
      }
    ],
    "static": false
  }
}
```

Puede asignar las funciones de dominio de Amazon OpenSearch Service a las funciones de IAM mediante la asignación de funciones de backend, como se muestra en el siguiente ejemplo:

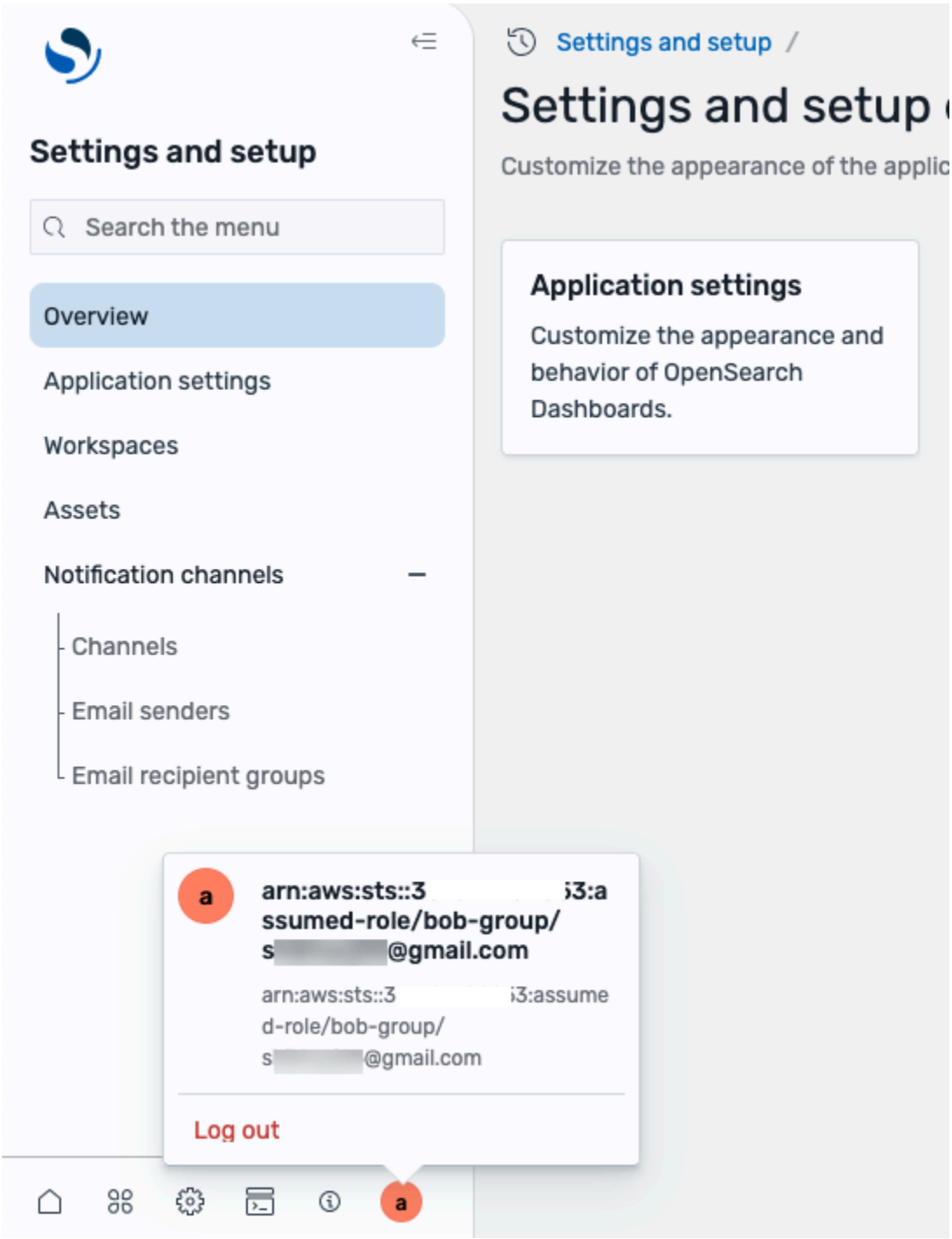
```
{
  "bob-group": {
    "hosts": [],
    "users": [],
    "reserved": false,
    "hidden": false,
    "backend_roles": [
      "arn:aws:iam::111222333444:role/bob-group"
    ],
    "and_backend_roles": []
  },
  "alice-group": {
    "hosts": [],
    "users": [],
    "reserved": false,
```

```
    "hidden": false,
    "backend_roles": [
      "arn:aws:iam::111222333444:role/alice-group"
    ],
    "and_backend_roles": []
  }
}
```

Paso 4: Compruebe la experiencia de inicio de sesión único iniciada por el proveedor de identidad con SAML

Abra la URL del estado de retransmisión predeterminado para abrir la página de autenticación de Okta. Introduzca las credenciales de SC-end-user. Se lo redirigirá automáticamente a OpenSearch UI.

Para comprobar sus credenciales actuales, seleccione el ícono de usuario situado en la parte inferior del panel de navegación, como se muestra en la siguiente imagen:



También puede verificar los permisos de control de acceso detallados del usuario accediendo a las herramientas para desarrolladores en la parte inferior del panel de navegación y ejecutando consultas en la consola. A continuación, se muestran algunas consultas de ejemplo:

Example 1: Displays information about the current user

Solicitud:

```
GET _plugins/_security/api/account
```

Resultado:

```
{
  "user_name": "arn:aws:iam::XXXXXXXXXXXX:role/bob-group",
  "is_reserved": false,
  "is_hidden": false,
  "is_internal_user": false,
  "user_requested_tenant": null,
  "backend_roles": [
    "arn:aws:iam::XXXXXXXXXXXX:role/bob-group"
  ],
  "custom_attribute_names": [],
  "tenants": {
    "global_tenant": true,
    "arn:aws:iam::XXXXXXXXXXXX:role/bob-group": true
  },
  "roles": [
    "bob-group"
  ]
}
```

Example 2: Displays actions permitted for a user

Solicitud:

```
GET bob-test/_search
```

Resultado:

```
{
  "took": 390,
  "timed_out": false,
  "_shards": {
    "total": 5,
    "successful": 5,
    "skipped": 0,
    "failed": 0
  },
  "hits": {
```

```
    "total": {
      "value": 1,
      "relation": "eq"
    },
    "max_score": 1,
    "hits": [
      {
        "_index": "bob-test",
        "_id": "ui01N5UBCIHpjO8Jlvfy",
        "_score": 1,
        "_source": {
          "title": "Your Name",
          "year": "2016"
        }
      }
    ]
  }
}
```

Example 3: Displays actions not permitted for a user

Solicitud:

```
GET alice-test
```

Resultado:

```
{
  "error": {
    "root_cause": [
      {
        "type": "security_exception",
        "reason": "no permissions for [indices:admin/get]
 and User [name=arn:aws:iam::111222333444:role/bob-group,
 backend_roles=[arn:aws:iam::111222333444:role/bob-group], requestedTenant=null]"
      }
    ],
    "type": "security_exception",
    "reason": "no permissions for [indices:admin/get]
 and User [name=arn:aws:iam::111222333444:role/bob-group,
 backend_roles=[arn:aws:iam::111222333444:role/bob-group], requestedTenant=null]"
  },
  "status": 403
```

```
}
```

Paso 5: Configuración del control de acceso detallado basado en atributos de SAML

Con Amazon OpenSearch Service, puede utilizar un control de acceso detallado con SAML para asignar los usuarios y grupos de su proveedor de identidad a los usuarios y roles de control de acceso detallados de OpenSearch de forma dinámica. Puede asignar estas funciones a dominios específicos de OpenSearch y colecciones sin servidor y definir los permisos a nivel de índice y la seguridad a nivel de los documentos.

> **Note**
>
> Para obtener más información, consulte Enabling fine-grained access control, lea the section called “Control de acceso detallado”.

Temas

- Atributos SAML para el control de acceso preciso
- Tarea 1: Configuración de Okta para el control de acceso preciso
- Tarea 2: Configurar SAML en el dominio de OpenSearch
- Tarea 3: Configurar SAML en las colecciones de OpenSearch sin servidor

Atributos SAML para el control de acceso preciso

subjectKey

Se asigna a un atributo de usuario único, como el correo electrónico o el nombre de usuario, que identifica al usuario para la autenticación.

rolesKey

Se asigna a los atributos de grupo o rol de su IdP que determinan los roles o permisos de autorización.

Tarea 1: Configuración de Okta para el control de acceso preciso

Para configurar Okta para el control de acceso preciso

1. Agregue un nuevo atributo para la entidad principal de usuario de OpenSearch en la sección Instrucciones de atributos:

   - Nombre: `UserName`
   - Valor: `${user-email}`

   Este atributo se utiliza como clave de asunto en la configuración detallada de control de acceso de OpenSearch para la autenticación.
2. Agregue un atributo de grupo para los roles en la sección sentencia de atributos de grupo:

   - Nombre: `groups`
   - Filtro: `0penSearch_xxx`

   Este atributo se utiliza como clave de función para asignar grupos a funciones de control de acceso específicas de OpenSearch para su autorización.

Tarea 2: Configurar SAML en el dominio de OpenSearch

Para configurar SAML en el dominio OpenSearch

1. En la Consola de administración de AWS, identifique el dominio de OpenSearch Service para el que desea habilitar un control de acceso detallado destinado a los usuarios de OpenSearch UI.
2. Vaya a la página de detalles del dominio específico.
3. Seleccione la pestaña de configuración de seguridad y haga clic en Editar.
4. Amplíe SAML mediante IAM Federate.
5. Introduzca el `subjectKey` y `roleKey` que definió en Okta.
6. Seleccione Guardar cambios.

También puede configurar un control de acceso detallado mediante AWS CLI.

```
aws opensearch create-domain \
```

```
--domain-name testDomain \
--engine-version OpenSearch_1.3 \
--cluster-config
 InstanceType=r5.xlarge.search,InstanceCount=1,DedicatedMasterEnabled=false,ZoneAwarenessEnable
 \
--access-policies '{"Version": "2012-10-17",       "Statement":
[{"Effect":"Allow","Principal":{"AWS":"*"},"Action":"es:*","Resource":"arn:aws:es:us-
east-1:12345678901:domain/neosaml10/*"}]}' \
--domain-endpoint-options '{"EnforceHTTPS":true,"TLSSecurityPolicy":"Policy-Min-
TLS-1-2-2019-07"}' \
--node-to-node-encryption-options '{"Enabled":true}' \
--encryption-at-rest-options '{"Enabled":true}' \
--advanced-security-options
 '{"Enabled":true,"InternalUserDatabaseEnabled":true,"MasterUserOptions":
{"MasterUserName":"********","MasterUserPassword":"********"}, "IAMFederationOptions":
{"Enabled": true,"SubjectKey":"TestSubjectKey","RolesKey":"TestRolesKey"}}' \
--ebs-options "EBSEnabled=true,VolumeType=gp2,VolumeSize=300" \
--no-verify-ssl \
--endpoint-url https://es.us-east-1.amazonaws.com \
--region us-east-1
```

Para actualizar un dominio existente:

```
aws opensearch update-domain-config \
--domain-name testDomain \
--advanced-security-options
 '{"Enabled":true,"InternalUserDatabaseEnabled":true,"MasterUserOptions":
{"MasterUserName":"********","MasterUserPassword":"********"}, "IAMFederationOptions":
{"Enabled": true,"SubjectKey":"TestSubjectKey","RolesKey":"TestRolesKey"}}' \
--ebs-options "EBSEnabled=true,VolumeType=gp2,VolumeSize=300" \
--no-verify-ssl \
--endpoint-url https://es.us-east-1.amazonaws.com \
--region us-east-1
```

Tarea 3: Configurar SAML en las colecciones de OpenSearch sin servidor

Para configurar un control de acceso detallado basado en SAML en OpenSearch sin servidor

1. Abra la Consola de administración de AWS y vaya a Amazon OpenSearch Service.
2. En el panel de navegación, en Sin servidor, seleccione Seguridad y, a continuación, seleccione Autenticación.
3. En la sección Federación de IAM, seleccione Editar.

Puede controlar el control de acceso específico basado en atributos de SAML mediante esta configuración. La federación de IAM está desactivada de forma predeterminada.

4. Seleccione Activar la federación de IAM.
5. Introduzca los valores `subjectKey` y `roleKey` que definió en Okta.

   Para obtener más información, consulte the section called “Atributos SAML para el control de acceso preciso ”.
6. Seleccione Guardar.
7. En el panel de navegación, en Sin servidor, elija Política de acceso a datos.
8. Actualice una política existente o cree una nueva.
9. Amplíe una regla, elija Agregar entidades principales y, a continuación, seleccione Usuarios y grupos federados de IAM.
10. Agregue las entidades principales necesarias y seleccione Guardar.
11. Elija Conceder.
12. Bajo esta regla, haga lo siguiente:

    - Seleccione los permisos que desee definir para las entidades principales seleccionadas.
    - Especifique las colecciones a las que desea aplicar los permisos.
    - Si lo desea, defina los permisos a nivel de índice.

    > **Note**
    >
    > Puede crear varias reglas para asignar diferentes permisos a diferentes grupos de entidades principales.
13. Cuando termine, elija Guardar.
14. Elija Crear.

Como alternativa, puede usar CLI para crear las configuraciones de seguridad para las colecciones, como se indica a continuación:

```
aws opensearchserverless create-security-config --region "region"  --type iamfederation
 --name "configuration_name" --description "description" --iam-federation-options
 '{"groupAttribute":"GroupKey","userAttribute":"UserKey"}'
```

Administración de las asociaciones de orígenes de datos y los permisos de acceso a la nube privada virtual

Utilice los procedimientos de esta sección para administrar las asociaciones de orígenes de datos y configurar los permisos de acceso necesarios para una nube privada virtual (VPC).

Temas

- Asociación de un origen de datos a una aplicación de OpenSearch UI
- Administrar el acceso a los dominios en una VPC
- Configuración del acceso a las colecciones de OpenSearch sin servidor en una VPC

Asociación de un origen de datos a una aplicación de OpenSearch UI

Tras crear una aplicación de OpenSearch UI, puede utilizar la consola o la AWS CLI para asociarla a uno o más orígenes de datos. Después de esto, los usuarios finales pueden recuperar los datos de estos orígenes de datos para buscarlos, trabajar con los paneles, etc.

Asociar un origen de datos a una aplicación de OpenSearch UI (consola)

Para asociar un origen de datos a una aplicación de OpenSearch UI mediante la consola

1. Inicie sesión en la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. Elija OpenSearch UI (Dashboards) y, a continuación, elija el nombre de una aplicación de OpenSearch UI.
3. En el área Orígenes de datos asociados, seleccione Administrar orígenes de datos.
4. Elija entre los dominios y colecciones de OpenSearch que desee asociar a la aplicación.

> **Tip**
>
> Si no encuentra los orígenes de datos que busca, póngase en contacto con sus administradores para que le concedan los permisos necesarios. Para obtener más información, consulte the section called “Permisos para crear una aplicación que utilice la autenticación del Centro de identidades de IAM (opcional)”.

5. Elija Siguiente y, a continuación, elija Guardar.

Una vez que haya asociado un origen de datos a la aplicación, se habilitará el botón Lanzar aplicación en la página de detalles de la aplicación. Puede seleccionar Iniciar aplicación para abrir la página de bienvenida a OpenSearch, en la que puede crear y administrar espacios de trabajo.

Para obtener más información sobre cómo trabajar con WorkSpaces, consulte the section called "Uso de los espacios de trabajo de Amazon OpenSearch Service".

Administrar el acceso a los dominios en una VPC

Si un dominio de OpenSearch en una VPC estaba asociado a la aplicación, el administrador de la VPC debe autorizar el acceso entre OpenSearch UI y la VPC mediante la consola o la AWS CLI.

Administrar el acceso a los dominios en una VPC (consola)

Para configurar el acceso a un dominio de VPC mediante Consola de administración de AWS:

1. Inicie sesión en la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. En el panel de navegación de la izquierda, seleccione Dominios y, a continuación, elija el nombre del dominio de VPC.

   -o bien-

   Seleccione Crear dominio y, a continuación, configure los detalles del dominio.
3. Elija la pestaña Puntos de conexión de VPC y, a continuación, elija Autorizar entidad principal.
4. En el cuadro de diálogo Autorizar entidades principales, seleccione Autorizar entidades principales de otros servicios de AWS y, a continuación, seleccione las aplicaciones de OpenSearch (panel de control) de la lista.
5. Seleccione Autorizar.

Administrar el acceso a los dominios en una VPC (AWS CLI)

Para autorizar un dominio de VPC mediante la AWS CLI

Para autorizar el dominio VPC mediante la AWS CLI, ejecute el siguiente comando. Reemplace los *valores de marcador de posición* con su información.

```
aws opensearch authorize-vpc-endpoint-access \
```

```
--domain-name domain-name \
--service application.opensearchservice.amazonaws.com \
--region region-id
```

Para revocar una asociación de dominio de VPC mediante la consola

Cuando ya no se necesita una asociación, el propietario del dominio de VPC puede revocar el acceso mediante el siguiente procedimiento.

1. Inicie sesión en la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. En el panel de navegación de la izquierda, seleccione Dominios y, a continuación, elija el nombre del dominio de VPC.
3. Elija la pestaña de puntos de conexión de VPC y, a continuación, seleccione el botón de la fila de aplicaciones de OpenSearch (panel de control).
4. Elija Revocar acceso.

Cómo revocar una asociación de dominio de VPC mediante AWS CLI

Para revocar una asociación de dominio de VPC con la aplicación OpenSearch UI, use el siguiente comando. Reemplace los *valores de marcador de posición* con su información.

```
aws opensearch revoke-vpc-endpoint-access \
    --domain-name domain-name \
    --service application.opensearchservice.amazonaws.com \
    --region region-id
```

Configuración del acceso a las colecciones de OpenSearch sin servidor en una VPC

Si se asoció a la aplicación una colección Amazon OpenSearch sin servidor en una VPC, el administrador de la VPC puede autorizar el acceso si crea una nueva política de red y la adjunta a la colección.

Configuración del acceso a las colecciones de OpenSearch sin servidor en una VPC (consola)

Para configurar el acceso a las colecciones OpenSearch sin servidor en una VPC mediante la consola

1. Inicie sesión en la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. En el panel de navegación de la izquierda, seleccione Políticas de red, elija el nombre de la política de red y, a continuación, seleccione Editar.

   -o bien-

   Elija Crear política de red y, a continuación, configure los detalles de la política.
3. En el área Tipo de acceso, seleccione Privado (recomendado) y, a continuación, seleccione Acceso privado al servicio de AWS.
4. En el campo de búsqueda, seleccione Servicio y, a continuación, seleccione `application.opensearchservice.amazonaws.com`.
5. En el área Tipo de recurso, seleccione la casilla Habilitar el acceso al punto de conexión de OpenSearch.
6. Para buscar colecciones o introducir términos de prefijo específicos, en el campo de búsqueda, seleccione Nombre de la colección y, a continuación, introduzca o seleccione el nombre de las colecciones para asociarlas a la política de red.
7. Seleccione Crear para una nueva política de red o Actualizar para una política de red existente.

Configuración del acceso a las colecciones de OpenSearch sin servidor en una VPC (AWS CLI)

Para configurar el acceso a las colecciones de OpenSearch sin servidor en una VPC mediante la AWS CLI

1. Cree una política con JSON similar a la siguiente. Reemplace los *valores de marcador de posición* con su información.

```
{
    "Description" : "policy-description",
    "Rules": [{
```

```
       "ResourceType" : "collection",
        "Resource" : ["collection/collection_name"]
     }],
    "SourceServices" : [
          "application.opensearchservice.amazonaws.com"
      ],
      "AllowFromPublic" : false
}
```

2. Cree o actualice una política de red para que una colección de una VPC funcione con las aplicaciones de OpenSearch UI.

   Create a network policy

   Ejecute el siguiente comando. Reemplace los *valores de marcador de posición* con su información.

   ```
   aws opensearchserverless create-security-policy \
       --type network  \
       --region region \
       --endpoint-url endpoint-url \
       --name network-policy-name \
       --policy file:/path_to_network_policy_json_file
   ```

   El comando devuelve información similar a la siguiente:

   ```
   {
       "securityPolicyDetail": {
           "createdDate": ******,
           "lastModifiedDate": ******,
           "name": "network-policy-name",
           "policy": [
               {
                   "SourceVPCEs": [],
                   "AllowFromPublic": false,
                   "Description": "",
                   "Rules": [
                       {
                           "Resource": [
                               "collection/network-policy-name"
                           ],
                           "ResourceType": "collection"
   ```

```
                    }
                ],
                "SourceServices": [
                    "application.opensearchservice.amazonaws.com"
                ]
            }
        ],
        "policyVersion": "******",
        "type": "network"
    }
}
```

Update a network policy

Ejecute el siguiente comando. Reemplace los *valores de marcador de posición* con su información.

```
aws opensearchserverless update-security-policy \
    --type network  \
    --region region \
    --endpoint-url endpoint-url \
    --name network-policy-name \
    --policy-version "policy_version_from_output_of_network_policy_creation" \
    --policy file:/path_to_network_policy_json_file
```

El comando devuelve información similar a la siguiente:

```
{
    "securityPolicyDetail": {
        "createdDate": ******,
        "lastModifiedDate": ******,
        "name": "network-policy-name",
        "policy": [
            {
                "SourceVPCEs": [],
                "AllowFromPublic": false,
                "Description": "",
                "Rules": [
                    {
                        "Resource": [
                            "collection/network-policy-name"
                        ],
```

```
                        "ResourceType": "collection"
                    }
                ],
                "SourceServices": [
                    "application.opensearchservice.amazonaws.com"
                ]
            }
        ],
        "policyVersion": "******",
        "type": "network"
    }
}
```

Uso de los espacios de trabajo de Amazon OpenSearch Service

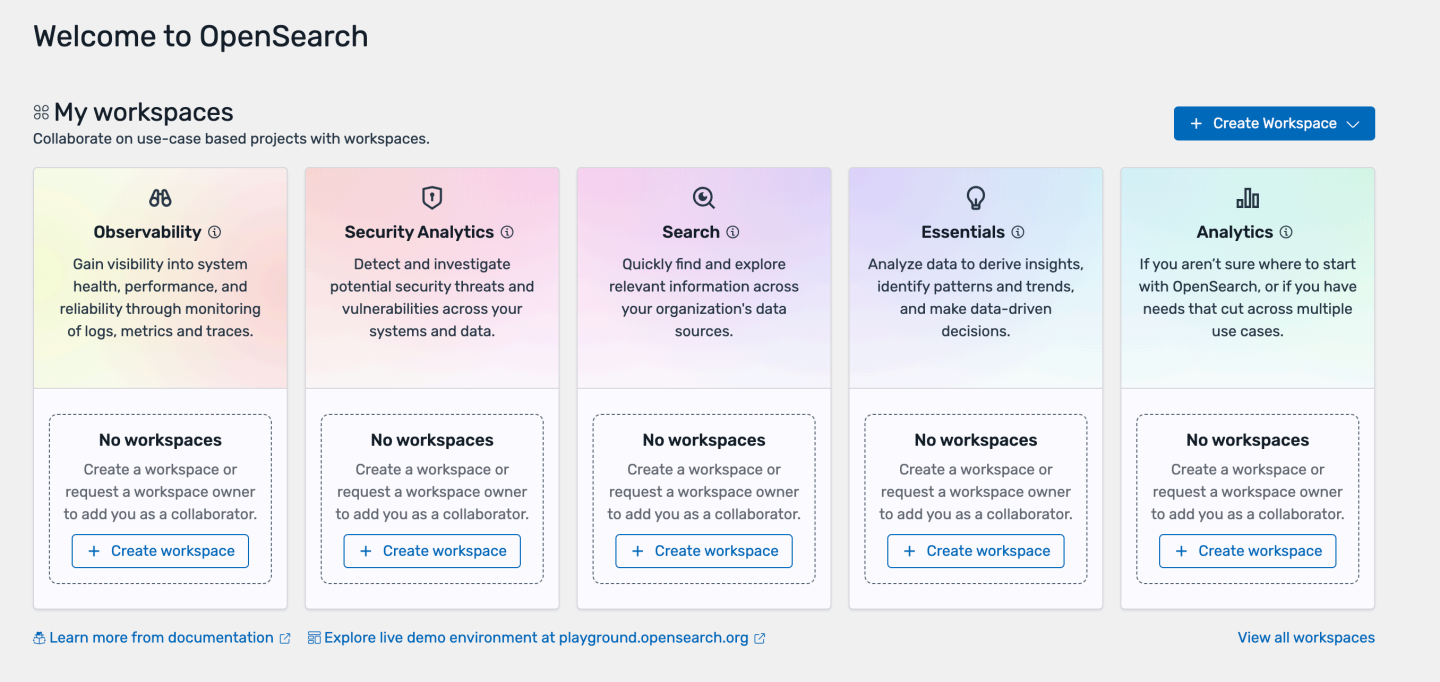
Amazon OpenSearch Service permite crear varios espacios de trabajo para casos de uso específicos. Cada espacio de trabajo ofrece una experiencia personalizada para casos de uso populares, como la observabilidad, el análisis de seguridad y la búsqueda. El espacio de trabajo también admite la administración de colaboradores, de modo que puede compartir su espacio de trabajo únicamente con los colaboradores con los que vaya a colaborar y administrar los permisos de cada uno de ellos.

Creación de espacios de trabajo para aplicaciones de OpenSearch UI

Una vez que se haya creado una aplicación de OpenSearch UI, que esta se haya asociado a los orígenes de datos y que se hayan configurado los permisos de usuario para la aplicación, puede lanzar la aplicación de OpenSearch UI para crear espacios de trabajo.

Para empezar a crear un espacio de trabajo, puede seleccionar el botón Abrir la aplicación en la página de detalles de la aplicación o utilizar la URL de la aplicación OpenSearch UI para abrir la página principal de la aplicación OpenSearch UI en una nueva ventana del navegador.

La aplicación de interfaz de usuario OpenSearch ofrece opciones para crear espacios de trabajo y enumera todos los espacios de trabajo existentes en la página de inicio, clasificados por caso de uso.



Para obtener más información sobre los tipos de instancia admitidos, consulte the section called "Tipos de WorkSpace".

La privacidad del espacio de trabajo y los colaboradores

Puede definir una configuración de privacidad para un espacio de trabajo como nivel de permiso predeterminado para todos los usuarios. Puede hacerlo mientras crea un espacio de trabajo o modifica un espacio de trabajo existente (en la pestaña Colaboradores del espacio de trabajo). Existen tres opciones entre las que elegir:

- Privado para los colaboradores: solo los colaboradores que agregue explícitamente al espacio de trabajo pueden acceder al espacio de trabajo. Puede definir niveles de permisos para cada colaborador.
- Cualquiera puede verlo: cualquier persona que tenga acceso a la aplicación de interfaz de usuario OpenSearch puede acceder al espacio de trabajo y ver sus activos, pero no puede realizar ningún cambio en el espacio de trabajo.
- Cualquier persona puede editar: cualquier persona que tenga acceso a la aplicación de interfaz de usuario OpenSearch puede acceder al espacio de trabajo, ver los activos que contiene y realizar cambios en los activos del espacio de trabajo.

En la pestaña Colaboradores del espacio de trabajo, puede agregar usuarios o roles de IAM y usuarios de AWS IAM Identity Center como colaboradores en un espacio de trabajo. Existen tres niveles de permisos entre los que elegir los colaboradores:

- Solo lectura: el colaborador solo puede ver los activos del espacio de trabajo. Esta configuración se anula si el espacio de trabajo está configurado para usar la configuración de privacidad Cualquiera puede editar.
- Leer y escribir: el colaborador puede ver y editar los activos del espacio de trabajo. Si el espacio de trabajo está configurado para usar la configuración de privacidad Todos pueden ver, el colaborador podrá seguir editando.
- Administrador: el colaborador puede actualizar la configuración y eliminar el espacio de trabajo. El colaborador también puede cambiar la configuración de privacidad del espacio de trabajo y administrar a los colaboradores. Al usuario que crea el espacio de trabajo se le asigna automáticamente la función de administrador del espacio de trabajo.

Tipos de WorkSpace

Amazon OpenSearch Service ofrece cinco tipos de espacios de trabajo, cada uno con características diferentes para los distintos casos de uso:

- El espacio de trabajo de observabilidad está diseñado para obtener visibilidad del estado, el rendimiento y la fiabilidad del sistema mediante la supervisión de los registros, las métricas y los rastreos.
- El espacio de trabajo de análisis de seguridad está diseñado para detectar e investigar posibles amenazas y vulnerabilidades de seguridad en sus sistemas y datos.
- El espacio de trabajo Search está diseñado para buscar y explorar rápidamente la información relevante en los orígenes de datos de su organización.
- El espacio de trabajo Essentials está diseñado para OpenSearch sin servidor como origen de datos y permite analizar los datos para obtener información, identificar patrones y tendencias y tomar decisiones basadas en los datos con rapidez. Puede buscar y explorar la información relevante de los orígenes de datos de su organización en un espacio de trabajo Essentials.
- El espacio de trabajo análisis (todas las características) está diseñado para casos de uso polivalentes y es compatible con todas las funciones disponibles en la interfaz de usuario (Dashboards) de OpenSearch Service.

Acceso a datos entre regiones y entre cuentas con búsqueda entre clústeres

La búsqueda entre clústeres en Amazon OpenSearch sin servidor permite realizar consultas y agregaciones en varios dominios conectados.

La búsqueda entre clústeres en Amazon OpenSearch sin servidor utiliza los conceptos de dominio de origen y dominio de destino. Una petición de búsqueda en clústeres se origina en un dominio de origen. El dominio de destino puede estar en una Cuenta de AWS o Región de AWS diferente (o en ambas) al dominio de origen desde el que se realiza la consulta. Mediante la búsqueda entre clústeres, puede configurar un dominio de origen para asociarlo a OpenSearch UI en la misma cuenta y, a continuación, crear conexiones a los dominios de destino. Como resultado, puede utilizar OpenSearch UI con los datos de los dominios de destino, incluso si se encuentran en una cuenta o región diferente.

Usted paga los cargos de transferencia de datos de AWS estándar por los datos transferidos dentro y fuera de Amazon OpenSearch Service. No se le cobrará por los datos transferidos entre los nodos de su dominio de OpenSearch Service. Para obtener más información acerca de los cargos de entrada y salida, consulte Transferencia de datos en la página Precios bajo demanda de Amazon EC2.

Puede usar la búsqueda entre clústeres como mecanismo para que OpenSearch UI se asocie a clústeres de una cuenta o región diferente. Todas las solicitudes de búsqueda en clústeres entre dominios se cifran en tránsito de forma predeterminada como parte del cifrado de nodo a nodo.

> **Note**
>
> OpenSearch de código abierto también tiene documentación para la búsqueda entre clústeres. La configuración difiere considerablemente entre los clústeres de código abierto en comparación con los dominios administrados de Amazon OpenSearch sin servidor.
> En concreto, en Amazon OpenSearch sin servidor, las conexiones entre clústeres se configuran mediante la Consola de administración de AWS en lugar de utilizar solicitudes `cURL`. Además, el servicio administrado utiliza AWS Identity and Access Management (IAM) para la autenticación entre clústeres, además de un control de acceso detallado.
> Por lo tanto, recomendamos utilizar esta documentación, en lugar de la documentación de OpenSearch de código abierto, para configurar la búsqueda entre clústeres para sus dominios.

Diferencias funcionales al utilizar la búsqueda entre clústeres

En comparación con los dominios normales, los dominios de destino creados mediante la búsqueda entre clústeres tienen las siguientes diferencias y requisitos funcionales:

- No puede escribir ni ejecutar comandos PUT en el clúster remoto. El acceso al clúster remoto es solo de lectura.
- Ambos dominios de origen y destino deben ser dominios de OpenSearch. No puede conectar un dominio de Elasticsearch ni clústeres de OpenSearch o Elasticsearch autoadministrados para OpenSearch UI.
- Un dominio puede tener un máximo de 20 conexiones a otros dominios. Esto incluye las conexiones entrantes y salientes.
- El dominio de origen debe estar en la misma versión o superior a la del dominio de destino. Si desea configurar conexiones bidireccionales entre dos dominios, los dos dominios deben estar en la misma versión. Recomendamos actualizar ambos dominios a la última versión antes de realizar la conexión. Si necesita actualizar los dominios después de configurar la conexión bidireccional, primero debe eliminar la conexión y, a continuación, volver a crearla.
- No puede usar diccionarios personalizados o SQL con la búsqueda en clústeres.
- No puede usar CloudFormation para conectar dominios.
- No se puede utilizar la búsqueda entre clústeres en instancias M3 o bursátiles (T2 y T3).
- La búsqueda entre clústeres no funciona en las colecciones de Amazon OpenSearch sin servidor.

Requisitos previos de búsqueda entre clústeres de OpenSearch UI

Antes de configurar la búsqueda entre clústeres con dos dominios de OpenSearch, asegúrese de que sus dominios cumplan con los siguientes requisitos:

- Control de acceso detallado habilitado para ambos dominios
- El cifrado de nodo a nodo está habilitado para ambos dominios

Temas

- Configuración de permisos de acceso para el acceso a datos entre regiones y cuentas mediante la búsqueda entre clústeres
- Crear una conexión entre dominios

- Probar su configuración de seguridad para el acceso a datos entre cuentas y regiones con búsqueda entre clústeres
- Eliminación de una conexión

Configuración de permisos de acceso para el acceso a datos entre regiones y cuentas mediante la búsqueda entre clústeres

Al enviar una petición de búsqueda en clústeres al dominio de origen, el dominio evalúa esa petición en comparación con la política de acceso al dominio. La búsqueda en clústeres requiere un control de acceso preciso. A continuación, se muestra un ejemplo con una política de acceso abierto en el dominio de origen.

JSON

```
{
"Version":"2012-10-17",
"Statement": [
    {
        "Effect": "Allow",
        "Principal": {
          "AWS": [
            "*"
          ]
      },
      "Action": [
        "es:ESHttp*"
      ],
      "Resource": "arn:aws:es:us-east-1:111222333444:domain/src-domain/*"
    }
  ]
}
```

> **Note**
>
> Si incluye índices remotos en la ruta, debe codificar la URL del URI en el ARN del dominio. Por ejemplo, use el siguiente formato de ARN: `:arn:aws:es:us-east-1:111222333444:domain/my-domain/local_index,dst%3Aremote_index`

No use el siguiente formato ARN. `arn:aws:es:us-east-1:111222333444:domain/my-domain/local_index,dst:remote_index.`

Si elige utilizar una política de acceso restrictivo además de un control de acceso detallado, la política debe permitir el acceso a `es:ESHttpGet` como mínimo. A continuación, se muestra un ejemplo:

JSON

```
{
"Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "arn:aws:iam::111222333444:user/john-doe"
        ]
      },
      "Action": "es:ESHttpGet",
      "Resource": "arn:aws:es:us-east-1:111122223333:domain/my-domain/*"
    }
  ]
}
```

El control de acceso detallado del dominio de origen evalúa la solicitud para determinar si está firmada con credenciales básicas de IAM o HTTP válidas. Si es así, el control de acceso detallado evalúa a continuación si el usuario tiene permiso para realizar la búsqueda y acceder a los datos.

Los siguientes son los requisitos de permiso para las búsquedas:

- Si la solicitud solo busca datos en el dominio de destino (por ejemplo, `dest-alias:dest-index/_search`)), solo necesitará permisos en el dominio de destino.
- Si la solicitud busca datos en ambos dominios (por ejemplo, `source-index,dest-alias:dest-index/_search`)), necesita permisos en ambos dominios.

- En el control de acceso detallado, los usuarios deben tener el permiso `indices:admin/shards/search_shards` además de los permisos de lectura o estándar para los índices pertinentes.

El dominio de origen pasa la solicitud al dominio de destino. El dominio de destino evalúa esta solicitud frente a su política de acceso al dominio. Para admitir todas las características de OpenSearch UI, como indexar documentos y realizar búsquedas estándar, se deben establecer permisos completos. A continuación, se muestra un ejemplo de nuestra política recomendada en el dominio de destino:

JSON

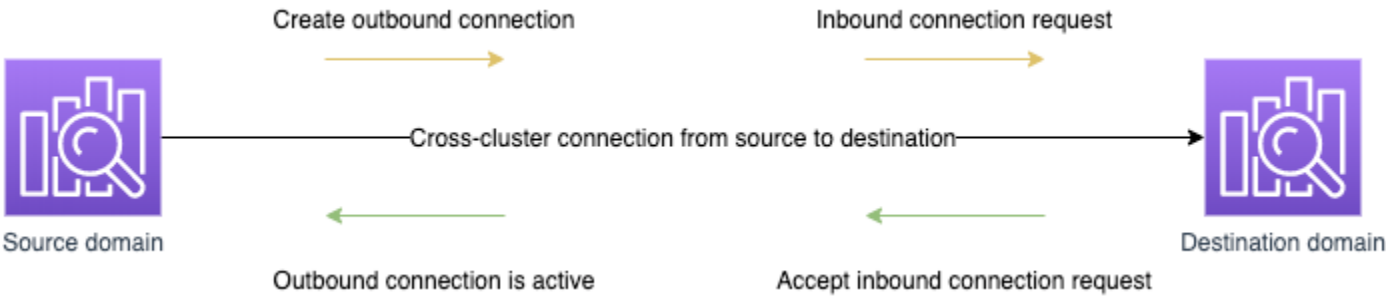
```
{
"Version":"2012-10-17",
 "Statement": [
    {
       "Effect": "Allow",
       "Principal": {
         "AWS": [
           "*"
        ]
      },
      "Action": [
        "es:ESHttp*"
      ],
      "Resource": "arn:aws:es:us-east-2:111222333444:domain/my-destination-
domain/*"
    },
    {
    "Effect": "Allow",
        "Principal": {
    "AWS": "*"
      },
      "Action": "es:ESCrossClusterGet",
      "Resource": "arn:aws:es:us-east-2:111222333444:domain/"
    }
  ]
}
```

Si solo desea realizar búsquedas básicas, el requisito mínimo de la política es que el permiso `es:ESCrossClusterGet` se solicite para el dominio de destino sin el uso de caracteres comodín. Por ejemplo, en la política anterior, especificaría el nombre de dominio como */my-destination-domain* y no */my-destination-domain/**.

En ese caso, el dominio de destino realiza la búsqueda y devuelve los resultados al dominio de origen. El dominio de origen combina sus propios resultados (si los hay) con los resultados del dominio de destino y los devuelve.

Crear una conexión entre dominios

Una conexión entre clústeres es unidireccional desde el dominio de origen hasta el dominio de destino. Esto significa que los dominios de destino (en una cuenta o región diferente) no pueden consultar el dominio de origen, que es local en OpenSearch UI. El dominio de origen crea una conexión outbound al dominio de destino. El dominio de destino recibe una solicitud de conexión inbound del dominio de origen.



Para crear una conexión entre dominios

1. Inicie sesión en la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. En el panel de navegación de la izquierda, seleccione Dominios.
3. Elija el nombre de un dominio que sirva como dominio de origen y, a continuación, elija la pestaña Conexiones.
4. En la sección Conexiones de salida, seleccione Solicitar.
5. En Alias de conexión, ingrese un nombre para la conexión. El alias de conexión se usa en OpenSearch UI para seleccionar los dominios de destino.
6. En el modo de conexión, elija Directo para realizar búsquedas o replicaciones entre clústeres.
7. Para especificar que la conexión debe omitir los clústeres no disponibles durante una búsqueda, active la casilla Omitir los clústeres no disponibles. Esta configuración garantiza que las

consultas entre clústeres devuelvan resultados parciales a pesar de que se produzcan errores en uno o más clústeres remotos.

8. En Clúster de destino, elija entre Conectarse a un clúster de esta Cuenta de AWS o Conectarse a un clúster de otra Cuenta de AWS.
9. En el ARN de dominio remoto, escriba el nombre de recurso de Amazon (ARN) para el clúster. El ARN del dominio se encuentra en el área de información general de la página de detalles del dominio.

   El dominio debe cumplir los siguientes requisitos:

   - El formato del ARN debe ser `arn:`*`partition`*`:es:`*`regionaccount-id`*`:`*`type`*`/`*`domain-id`*. Por ejemplo:

     ```
     arn:aws:es:us-east-2:111222333444:domain/my-domain
     ```
   - El dominio debe configurarse para usar OpenSearch versión 1.0 (o posterior) o Elasticsearch versión 6.7 (o posterior).
   - Debe estar habilitado el control de acceso preciso en el dominio.
   - El dominio debe ejecutar OpenSearch.
10. Seleccione Request (Solicitar).

La búsqueda entre clústeres valida primero la solicitud de conexión para asegurarse de que se cumplen los requisitos previos. Si los dominios son incompatibles, la solicitud de conexión entra en el estado `Validation failed`.

Una vez correctamente validada la solicitud de conexión, se envía al dominio de destino, donde tiene que aprobarse. Mientras no se obtenga la aprobación, la conexión permanecerá en el estado de `Pending acceptance`. Cuando se acepta la solicitud de conexión en el dominio de destino, el estado cambia a `Active` y el dominio de destino se vuelve disponible para consultas.

La página de dominio muestra el estado general del dominio y los detalles del estado de la instancia del dominio de destino. Los propietarios de dominios tienen la flexibilidad de crear, visualizar, eliminar y monitorear conexiones desde o hacia sus dominios.

Una vez establecida la conexión, se cifra todo el tráfico que circule entre los nodos de los dominios conectados. Si conecta un dominio de VPC a un dominio que no es VPC y el dominio que no es VPC es un punto de conexión público que puede recibir tráfico de Internet, el tráfico entre clúster entre los dominios sigue cifrado y seguro.

Probar su configuración de seguridad para el acceso a datos entre cuentas y regiones con búsqueda entre clústeres

Una vez que haya configurado los permisos de acceso a los datos entre regiones y cuentas mediante la búsqueda entre clústeres, le recomendamos que pruebe la configuración con Postman, una plataforma de terceros para el desarrollo colaborativo de las API.

Para configurar la configuración de seguridad mediante Postman

1. En el dominio de destino, indexe un documento: Lo que sigue es un ejemplo de resultado.

```
POST https://dst-domain.us-east-1.es.amazonaws.com/books/_doc/1
{
"Dracula": "Bram Stoker"
}
```

2. Para consultar este índice desde el dominio de origen, incluya el alias de conexión del dominio de destino dentro de la consulta. Puede encontrar el alias de conexión en la pestaña Conexiones del panel de control del dominio. A continuación, se muestra un ejemplo de solicitud y una respuesta truncada:

```
GET https://src-domain.us-east-1.es.amazonaws.com/connection_alias:books/_search
{
    ...
  "hits": [
    {
"_index": "source-destination:books",
      "_type": "_doc",
      "_id": "1",
      "_score": 1,
      "_source": {
"Dracula": "Bram Stoker"
      }
    }
  ]
}
```

3. (Opcional) Puede crear una configuración que incluya varios dominios en una sola búsqueda. Por ejemplo, supongamos que tiene lo siguiente:

   Una conexión entre `domain-a` a `domain-b`, con un alias de conexión denominado `cluster_b`

Una conexión entre `domain-a` a `domain-c`, con un alias de conexión denominado `cluster_c`

En este caso, las búsquedas incluyen el contenido `domain-a`, `domain-b` y `domain-c`. A continuación se muestra un ejemplo de una solicitud y la respuesta:

Solicitud

```
GET https://src-domain.us-east-1.es.amazonaws.com/
local_index,cluster_b:b_index,cluster_c:c_index/_search
{
  "query": {
    "match": {
      "user": "domino"
    }
  }
}
```

Respuesta:

```
{
"took": 150,
  "timed_out": false,
  "_shards": {
"total": 3,
    "successful": 3,
    "failed": 0,
    "skipped": 0
  },
  "_clusters": {
"total": 3,
    "successful": 3,
    "skipped": 0
  },
  "hits": {
"total": 3,
    "max_score": 1,
    "hits": [
      {
"_index": "local_index",
        "_type": "_doc",
        "_id": "0",
        "_score": 1,
```

```
        "_source": {
"user": "domino",
          "message": "This is message 1",
          "likes": 0
        }
      },
      {
"_index": "cluster_b:b_index",
        "_type": "_doc",
        "_id": "0",
        "_score": 2,
        "_source": {
"user": "domino",
          "message": "This is message 2",
          "likes": 0
        }
      },
      {
"_index": "cluster_c:c_index",
        "_type": "_doc",
        "_id": "0",
        "_score": 3,
        "_source": {
"user": "domino",
          "message": "This is message 3",
          "likes": 0
        }
      }
    ]
  }
}
```

Si no eligió omitir los clústeres no disponibles en la configuración de la conexión, todos los clústeres de destino que busca tienen que estar disponibles para que la petición de búsqueda se ejecute correctamente. De lo contrario, se produce un error en toda la solicitud; incluso si uno de los dominios no está disponible, la búsqueda no devuelve ningún resultado.

Eliminación de una conexión

La eliminación de una conexión detiene cualquier operación de búsqueda entre clústeres en el dominio de destino.

Puede realizar estos pasos en el dominio de origen o de destino para eliminar la conexión. Una vez quitada la conexión, seguirá visible con el estado `Deleted` durante un periodo de 15 días.

No se puede eliminar un dominio con conexiones activas entre clústeres. Para eliminar un dominio, primero quite todas las conexiones entrantes y salientes de ese dominio. Esto es para asegurarse de tener en cuenta a los usuarios del dominio entre clústeres antes de eliminar el dominio.

Para eliminar una conexión

1. Inicie sesión en la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. En el panel de navegación de la izquierda, seleccione Dominios.
3. Elija el nombre de un dominio y, a continuación, la pestaña Contenido.
4. El nombre de la conexión que se va a eliminar.
5. Elija Eliminar y, a continuación, confirme la eliminación.

Administración del acceso a OpenSearch UI desde un punto de conexión de VPC

Puede usar un AWS PrivateLink para crear una conexión privada entre la VPC y OpenSearch UI. Con esta conexión, puede acceder a las aplicaciones de OpenSearch Server como si estuvieran en la misma VPC. De esta forma, no necesita configurar una puerta de enlace de internet, un dispositivo NAT, una conexión VPN ni AWS Direct Connect para establecer la conexión. Las instancias de la VPC no necesitan direcciones IP públicas para acceder a Amazon OpenSearch UI.

Esta conexión privada se establece mediante la creación de un punto de conexión de interfaz alimentado por AWS PrivateLink. Se crea una interfaz de red de punto de conexión en cada subred especificada para el punto de conexión de interfaz. Se trata de interfaces de red administradas por el solicitante que sirven como punto de entrada para el tráfico destinado a aplicaciones de OpenSearch UI.

Creación de una conexión privada entre una VPC y OpenSearch UI

Puede crear una conexión privada para acceder a OpenSearch UI desde una VPC mediante la Consola de administración de AWS o la AWS CLI.

Creación de una conexión privada entre una VPC y OpenSearch UI (consola)

Para crear una conexión privada entre una VPC y OpenSearch UI mediante la consola

1. Inicie sesión en la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. En el panel de navegación de la izquierda, amplíe Sin servidor y seleccione Puntos de conexión de VPC.
3. Seleccione Crear punto de conexión de VPC.
4. En Nombre, ingrese un nombre para el punto de conexión.
5. Para VPC, seleccione la VPC desde la que accederá a las aplicaciones de OpenSearch UI.
6. En Subredes, seleccione la subred desde la que accederá a las aplicaciones de OpenSearch UI.

   > **Note**
   >
   > La dirección IP y el tipo de DNS del punto de conexión se basan en el tipo de subred:
   >
   > - Dualstack: si todas las subredes tienen rangos de direcciones IPv4 y IPv6.
   > - IPv6: si todas las subredes son subredes IPv6.
   > - IPv4: si todas las subredes tienen rangos de direcciones IPv4.

7. En Grupos de seguridad, seleccione el grupo de seguridad que se va a asociar con las interfaces de red de punto de conexión de VPC.

   > **Note**
   >
   > Este es un paso fundamental en el que limita los puertos, protocolos y orígenes para el tráfico entrante que autoriza para el punto de conexión. Asegúrese de que las reglas del grupo de seguridad permitan a los recursos que utilizarán los puntos de conexión de VPC comunicarse con OpenSearch UI para, a su vez, comunicarse con la interfaz de la red del punto de conexión.

8. 8. Seleccione Crear punto de conexión.

Creación de una conexión privada entre una VPC y OpenSearch UI (AWS CLI)

Para crear una conexión privada entre una VPC y OpenSearch UI mediante la AWS CLI

Ejecute el siguiente comando. Reemplace los *valores de marcador de posición* con su información.

```
aws opensearchserverless create-vpc-endpoint \
    --region region \
    --endpoint endpoint \
    --name vpc_endpoint_name \
    --vpc-id vpc_id \
    --subnet-ids subnet_ids
```

Actualización de la política de puntos de conexión de VPC para permitir el acceso a la aplicación de OpenSearch UI

Después de crear la conexión privada, actualice la política de puntos de conexión de VPC para permitir el acceso a la aplicación de OpenSearch UI en la política de puntos de conexión de VPC mediante la especificación del ID de la aplicación.

Para obtener más información sobre cómo actualizar la política de punto de conexión de VPC, consulte Actualización de una política de punto de conexión de VPC en la Guía de AWS PrivateLink.

Asegúrese de que la política de punto de conexión de VPC incluya la siguiente declaración. Reemplace el *valor de marcador de posición* con su propia información.

```
{
    "Statement": [{
        "Action": ["opensearch:*"],
        "Effect": "Allow",
        "Principal": "*",
        "Resource": "*",
        "Condition": {
            "StringEquals": {
                "opensearch:ApplicationId": ["opensearch-ui-application-id"]
            }
        }
    }]
}
```

Revocación del acceso a OpenSearch UI en una política de punto de conexión de VPC

OpenSearch UI requiere un permiso explícito en la política de punto de conexión de VPC para permitir a los usuarios acceder a la aplicación desde la VPC. Si ya no desea que los usuarios accedan a OpenSearch desde la VPC, puede eliminar el permiso en la política de punto de conexión. Después de esto, los usuarios reciben un mensaje de error `403 forbidden` al intentar acceder a OpenSearch UI.

Para obtener más información sobre cómo actualizar la política de punto de conexión de VPC, consulte Actualización de una política de punto de conexión de VPC en la Guía de AWS PrivateLink.

El siguiente es un ejemplo de una política de punto de conexión de VPC que deniega el acceso a las aplicaciones de la interfaz de usuario desde la VPC:

```
{
    "Statement": [{
        "Action": ["opensearch:*"],
        "Effect": "Allow",
        "Principal": "*",
        "Resource": "*",
        "Condition": {
            "StringEquals": {
                "opensearch:ApplicationId": [""]
            }
        }
    }]
}
```

Puntos de conexión y cuotas de OpenSearch UI

Para conectarse mediante programación a un servicio de AWS, utilice un punto de conexión. Las Service Quotas, que también se denominan límites, establecen el número máximo de recursos u operaciones de servicio que puede haber en una cuenta de AWS.

Amazon OpenSearch UI es la interfaz de usuario de OpenSearch de última generación para OpenSearch Dashboards. Proporciona puntos de conexión para acceder a sus paneles de OpenSearch. Utilice este tema para buscar los puntos de conexión y las Service Quotas de OpenSearch UI.

Para obtener más información sobre otros servicios de OpenSearch, consulte Puntos de conexión y cuotas del servicio.

Puntos de conexión de OpenSearch UI

OpenSearch UI está disponible en las siguientes regiones:

Nombre de la región	Región	Punto de conexión	Protocolo
Asia-Pacífico (Mumbai)	ap-south-1	opensearch.ap-south-1.amazonaws.com es.ap-south-1.amazonaws.com	HTTPS HTTPS
Europa (París)	eu-west-3	opensearch.eu-west-3.amazonaws.com es.eu-west-3.amazonaws.com	HTTPS HTTPS
Este de EE. UU. (Ohio)	us-east-2	opensearch.us-east-2.amazonaws.com es.us-east-2.amazonaws.com	HTTPS HTTPS
Europa (Irlanda)	eu-west-1	opensearch.eu-west-1.amazonaws.com es.eu-west-1.amazonaws.com	HTTPS HTTPS
Europa (Fráncfort)	eu-central-1	opensearch.eu-central-1.amazonaws.com es.eu-central-1.amazonaws.com	HTTPS HTTPS
América del Sur (São Paulo)	sa-east-1	opensearch.sa-east-1.amazonaws.com es.sa-east-1.amazonaws.com	HTTPS HTTPS

Nombre de la región	Región	Punto de conexión	Protocolo
Este de EE. UU. (Norte de Virginia)	us-east-1	opensearch.us-east-1.amazonaws.com es.us-east-1.amazonaws.com	HTTPS HTTPS
Europa (Londres)	eu-west-2	opensearch.eu-west-2.amazonaws.com es.eu-west-2.amazonaws.com	HTTPS HTTPS
Asia-Pacífico (Tokio)	ap-northeast-1	zocalo.ap-northeast-1.amazonaws.com es.ap-northeast-1.amazonaws.com	HTTPS HTTPS
Oeste de EE. UU. (Oregón)	us-west-2	opensearch.us-west-2.amazonaws.com es.us-west-2.amazonaws.com	HTTPS HTTPS
Asia-Pacífico (Singapur)	ap-southeast-1	opensearch.ap-southeast-1.amazonaws.com es.ap-southeast-1.amazonaws.com	HTTPS HTTPS
Asia-Pacífico (Sídney)	ap-southeast-2	opensearch.ap-southeast-2.amazonaws.com es.ap-southeast-2.amazonaws.com	HTTPS HTTPS
Canadá (centro)	ca-central-1	opensearch.ca-central-1.amazonaws.com es.ca-central-1.amazonaws.com	HTTPS HTTPS

Nombre de la región	Región	Punto de conexión	Protocolo
Europa (Estocolmo)	eu-north-1	opensearch.eu-north-1.amazonaws.com es.eu-north-1.amazonaws.com	HTTPS HTTPS
Asia-Pacífico (Hong Kong)	ap-east-1	opensearch.ap-east-1.amazonaws.com es.ap-east-1.amazonaws.com	HTTPS HTTPS
Asia-Pacífico (Seúl)	ap-northeast-2	opensearch.ap-northeast-2.amazonaws.com es.ap-northeast-2.amazonaws.com	HTTPS HTTPS
Asia-Pacífico (Osaka)	ap-northeast-3	opensearch.ap-northeast-3.amazonaws.com es.ap-northeast-3.amazonaws.com	HTTPS HTTPS
Asia-Pacífico (Hyderabad)	ap-south-2	opensearch.ap-south-2.amazonaws.com es.ap-south-2.amazonaws.com	HTTPS HTTPS
Europa (España)	eu-south-2	opensearch.eu-south-2.amazonaws.com es.eu-south-2.amazonaws.com	HTTPS HTTPS
Oeste de EE. UU. (Norte de California)	us-west-1	opensearch.us-west-1.amazonaws.com es.us-west-1.amazonaws.com	HTTPS HTTPS

Nombre de la región	Región	Punto de conexión	Protocolo
Europa (Zúrich)	eu-central-2	opensearch.eu-central-2.amazonaws.com es.eu-central-2.amazonaws.com	HTTPS HTTPS
Europa (Milán)	eu-south-1	opensearch.eu-south-1.amazonaws.com es.eu-south-1.amazonaws.com	HTTPS HTTPS

Service Quotas de OpenSearch UI

Su cuenta de AWS tiene las siguientes cuotas relacionadas con los recursos de OpenSearch UI.

Nombre	Valor predeterminado	Ajustable	Notas
Aplicaciones de OpenSearch UI por cuenta y región	30	Sí	El número máximo de aplicaciones de OpenSearch UI que puede crear por cuenta de por región. Puede aumentar el límite a 50 mediante la cuota de servicio y conseguir que se apruebe automáticamente. Para solicitar un límite superior, envía un ticket de soporte.

Administración de índices en Amazon OpenSearch Service

Después de agregar datos a Amazon OpenSearch Service, a menudo necesita reindexarlos, trabajar con alias de índice, trasladar un índice a un almacenamiento más rentable o eliminarlo por completo. Este capítulo trata sobre el almacenamiento UltraWarm, el almacenamiento en frío y la administración de estados de índice. Para obtener información sobre las API de índice de OpenSearch, consulte la documentación de OpenSearch.

Temas

- Almacenamiento UltraWarm para Amazon OpenSearch Service
- Almacenamiento en frío para Amazon OpenSearch Service
- Instancias optimizadas para OpenSearch (OR1) para dominios de Amazon OpenSearch Service
- Administración de estados de índice en Amazon OpenSearch Service
- Resumen de índices en Amazon OpenSearch Service con acumulaciones de índices
- Transformación de índices en Amazon OpenSearch Service
- Replicación entre clústeres de Amazon OpenSearch Service
- Migración de índices de Amazon OpenSearch Service mediante la reindexación remota
- Administración de datos de series temporales en Amazon OpenSearch Service con Data Streams

Almacenamiento UltraWarm para Amazon OpenSearch Service

UltraWarm proporciona una forma rentable de almacenar grandes cantidades de datos de solo lectura en Amazon OpenSearch Service. Los nodos de datos estándar utilizan almacenamiento "en caliente", que adopta la forma de almacenes de instancias o volúmenes de Amazon EBS asociados a cada nodo. El almacenamiento en caliente proporciona el rendimiento más rápido posible para indexar y buscar nuevos datos.

En lugar de almacenamiento asociado, los nodos UltraWarm utilizan Amazon S3 y una sofisticada solución de almacenamiento en caché para mejorar el rendimiento. Para los índices en los que no se está escribiendo de forma activa, que se consultan con menos frecuencia y de los que no se necesita el mismo rendimiento, UltraWarm ofrece costos significativamente inferiores por GiB de datos. Dado que los índices templados son de solo lectura a menos que se los restaure al almacenamiento caliente, UltraWarm es más adecuado para datos inmutables, tales como registros.

En OpenSearch, los índices templados se comportan como cualquier otro índice. Puede consultarlos mediante las mismas API o utilizarlos para crear visualizaciones en OpenSearch Dashboards.

Temas

- Requisitos previos
- Requisitos de almacenamiento UltraWarm y consideraciones de rendimiento
- Precios de UltraWarm
- Habilitar UltraWarm
- Migración de índices al almacenamiento UltraWarm
- Automatizar migraciones
- Ajuste de migración
- Cancelación de migraciones
- Listado de índices calientes y templados
- Devolución de índices templados al almacenamiento caliente
- Restauración de índices templados a partir de instantáneas
- Instantáneas manuales de índices templados
- Migración de índices templados al almacenamiento frío
- Prácticas recomendadas para índices KNN
- Deshabilitar UltraWarm

Requisitos previos

UltraWarm tiene algunos requisitos previos importantes:

- UltraWarm requiere OpenSearch o Elasticsearch 6.8 o posterior.
- Para utilizar el almacenamiento "warm", los dominios deben tener nodos maestros dedicados.
- Cuando se utiliza un dominio Multi-AZ con modo de espera, la cantidad de nodos templados debe ser un múltiplo de la cantidad de zonas de disponibilidad que se utilizan.
- Si su dominio utiliza un tipo de instancia T2 o T3 para los nodos de datos, no puede utilizar el almacenamiento templado.
- Si su índice usa k-NN aproximado (`"index.knn":true`), puede pasarlo a almacenamiento en caliente a partir de la versión 2.17 y posteriores. Los dominios de versiones anteriores a la 2.17

pueden actualizarse a la 2.17 para utilizar esta funcionalidad, pero los índices KNN creados en versiones anteriores a la 2.x no pueden migrar a UltraWarm.

- Si el dominio utiliza control de acceso preciso, los usuarios deben asignarse al rol de `ultrawarm_manager` en OpenSearch Dashboards para realizar llamadas a la API UltraWarm.

> **Note**
>
> El rol de `ultrawarm_manager` puede no estar definido en algunos dominios de OpenSearch Service preexistentes. Si no ve el rol en Dashboards, debe crearlo manualmente.

Configurar permisos

Si habilita UltraWarm en un dominio de OpenSearch Service preexistente, la opción `ultrawarm_manager` puede no definirse en el dominio. Los usuarios que no sean administradores deben estar asignados a este rol para poder administrar índices templados en los dominios mediante un control de acceso detallado. Para crear el rol `ultrawarm_manager` de forma manual, siga estos pasos:

1. En OpenSearch Dashboards, vaya a Seguridad y elija Permisos.
2. Seleccione Crear grupo de acciones y configure los siguientes grupos:

Nombre del grupo	Permisos
`ultrawarm_cluster`	• `cluster:admin/ultrawarm/migration/list` • `cluster:monitor/nodes/stats`
`ultrawarm_index_read`	• `indices:admin/ultrawarm/migration/get` • `indices:admin/get`
`ultrawarm_index_write`	• `indices:admin/ultrawarm/migration/warm` • `indices:admin/ultrawarm/migration/hot` • `indices:monitor/stats` • `indices:admin/ultrawarm/migration/cancel`

3. Seleccione Roles y, a continuación, Crear rol.
4. Asigne al rol el nombre ultrawarm_manager.
5. Para Permisos de clúster, seleccione `ultrawarm_cluster` y `cluster_monitor`.
6. Para Índice, escriba `*`.
7. Para Permisos de índice, seleccione `ultrawarm_index_read`, `ultrawarm_index_write`, y `indices_monitor`.
8. Seleccione Crear.
9. Después de crear el rol, debe mapearlo a cualquier rol de usuario o backend que administre los índices UltraWarm.

Requisitos de almacenamiento UltraWarm y consideraciones de rendimiento

Como se explica en the section called “Cálculo de requisitos de almacenamiento”, los datos del almacenamiento en caliente generan una sobrecarga significativa: réplicas, espacio reservado para Linux y espacio reservado para OpenSearch Service. Por ejemplo, una partición principal de 20 GiB con una partición de réplica requiere aproximadamente 58 GiB de almacenamiento en caliente.

Debido a que utiliza Amazon S3, UltraWarm no genera ninguno de estos gastos. Al calcular los requisitos de almacenamiento UltraWarm, solo se tiene en cuenta el tamaño de las particiones principales. La durabilidad de los datos en S3 elimina la necesidad de mantener réplicas y S3 abstrae (y hace innecesarias) las consideraciones de sistemas operativos o de servicio. Esa misma partición de 20 GiB requiere solo 20 GiB de almacenamiento templado. Si aprovisiona una instancia `ultrawarm1.large.search`, puede usar los 20 TiB de su almacenamiento máximo para particiones principales. Consulte the section called “Cuotas de almacenamiento UltraWarm” para obtener un resumen de los tipos de instancia y la cantidad máxima de almacenamiento que puede abordar cada uno.

Incluso con UltraWarm, recomendamos un tamaño máximo de partición de 50 GiB. El número de núcleos de CPU y cantidad de RAM asignada a cada tipo de instancia UltraWarm brinda una idea del número de particiones que pueden buscar simultáneamente. Tenga en cuenta que, aunque solo las particiones principales cuentan para el almacenamiento UltraWarm en S3, OpenSearch Dashboards y `_cat/indices` aún informan del tamaño del índice UltraWarm como el total de todas las particiones primarias y de réplica.

Por ejemplo, cada instancia de `ultrawarm1.medium.search` tiene dos núcleos de CPU y puede abordar hasta 1,5 TiB de almacenamiento en S3. Dos de estas instancias tienen una combinación de 3 TiB de almacenamiento, que funciona en aproximadamente 62 particiones si cada partición es de 50 GiB. Si una solicitud al clúster solo busca cuatro de estas particiones, el rendimiento puede ser excelente. Si la solicitud es amplia y busca los 62, es posible que los cuatro núcleos de CPU tengan dificultades para realizar la operación. Monitoree el `WarmCPUUtilization` y `WarmJVMMemoryPressure` de las métricas de UltraWarm para comprender cómo manejan las instancias sus cargas de trabajo.

Si realiza búsquedas amplias o frecuentes, considere dejar los índices en el almacenamiento caliente. Al igual que cualquier otra carga de trabajo de OpenSearch, el paso más importante para determinar si UltraWarm satisface sus necesidades es realizar pruebas representativas en el cliente utilizando un conjunto de datos realista.

Precios de UltraWarm

Con el almacenamiento en caliente, se paga lo que se aprovisiona. Algunas instancias requieren un volumen de Amazon EBS asociado, mientras que otras incluyen un almacén de instancias. Tanto si el almacenamiento está vacío como si está lleno, se paga el mismo precio.

Con el almacenamiento UltraWarm, se paga por lo que se utiliza. Una instancia `ultrawarm1.large.search` puede poner a disposición hasta 20 TiB de almacenamiento en S3. Sin embargo, si únicamente se almacena 1 TiB de datos, solo se le facturará 1 TiB de datos. Al igual que todos los demás tipos de nodos, también se paga una tarifa por horas por cada nodo UltraWarm. Para más información, consulte the section called “Precios”.

Habilitar UltraWarm

La consola es la forma más sencilla de crear un dominio que utiliza almacenamiento templado. Al crear el dominio, elija Habilitar nodos de datos en caliente y la cantidad de nodos en caliente que desee. El mismo proceso básico funciona en dominios existentes, siempre que cumplan los requisitos previos. Incluso después de que el estado del dominio cambie de Procesando a Activo, es posible que UltraWarm no esté disponible para su uso durante varias horas.

Cuando se utiliza un dominio Multi-AZ con modo de espera, la cantidad de nodos templados debe ser un múltiplo de la cantidad de zonas de disponibilidad. Para obtener más información, consulte the section called “Multi-AZ con modo de espera”.

También puede usar la AWS CLI o la API de configuración para habilitar UltraWarm, concretamente las opciones `WarmEnabled`, `WarmCount` y `WarmType` en `ClusterConfig`.

> **Note**
>
> Los dominios admiten un número máximo de nodos templados. Para más información, consulte the section called “Cuotas”.

Ejemplo de comando de la CLI

El siguiente comando de la AWS CLI crea un dominio con tres nodos de datos, tres nodos maestros dedicados, seis nodos calientes y control de acceso detallado habilitado:

```
aws opensearch create-domain \
  --domain-name my-domain \
  --engine-version Opensearch_1.0 \
  --cluster-config
 InstanceCount=3,InstanceType=r6g.large.search,DedicatedMasterEnabled=true,DedicatedMasterType=
 \
  --ebs-options EBSEnabled=true,VolumeType=gp2,VolumeSize=11 \
  --node-to-node-encryption-options Enabled=true \
  --encryption-at-rest-options Enabled=true \
  --domain-endpoint-options EnforceHTTPS=true,TLSSecurityPolicy=Policy-Min-
TLS-1-2-2019-07 \
  --advanced-security-options
 Enabled=true,InternalUserDatabaseEnabled=true,MasterUserOptions='{MasterUserName=master-
user,MasterUserPassword=master-password}' \
  --access-policies '{"Version": "2012-10-17",       "Statement":
[{"Effect":"Allow","Principal":{"AWS":["123456789012"]},"Action":
["es:*"],"Resource":"arn:aws:es:us-west-1:123456789012:domain/my-domain/*"}]}' \
  --region us-east-1
```

Para obtener información detallada, consulte la Referencia de comandos de AWS CLI.

Ejemplo de solicitud a la API de configuración

La siguiente solicitud a la API de configuración crea un dominio con tres nodos de datos, tres nodos maestros dedicados y seis nodos templados con control de acceso detallado habilitado y una política de acceso restrictiva:

```
POST https://es.us-east-2.amazonaws.com/2021-01-01/opensearch/domain
{
  "ClusterConfig": {
    "InstanceCount": 3,
    "InstanceType": "r6g.large.search",
    "DedicatedMasterEnabled": true,
    "DedicatedMasterType": "r6g.large.search",
    "DedicatedMasterCount": 3,
    "ZoneAwarenessEnabled": true,
    "ZoneAwarenessConfig": {
      "AvailabilityZoneCount": 3
    },
    "WarmEnabled": true,
    "WarmCount": 6,
    "WarmType": "ultrawarm1.medium.search"
  },
  "EBSOptions": {
    "EBSEnabled": true,
    "VolumeType": "gp2",
    "VolumeSize": 11
  },
  "EncryptionAtRestOptions": {
    "Enabled": true
  },
  "NodeToNodeEncryptionOptions": {
    "Enabled": true
  },
  "DomainEndpointOptions": {
    "EnforceHTTPS": true,
    "TLSSecurityPolicy": "Policy-Min-TLS-1-2-2019-07"
  },
   "AdvancedSecurityOptions": {
    "Enabled": true,
    "InternalUserDatabaseEnabled": true,
    "MasterUserOptions": {
      "MasterUserName": "master-user",
      "MasterUserPassword": "master-password"
    }
  },
  "EngineVersion": "Opensearch_1.0",
  "DomainName": "my-domain",
```

```
  "AccessPolicies": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":\"Allow
\",\"Principal\":{\"AWS\":[\"123456789012\"]},\"Action\":[\"es:*\"],\"Resource\":
\"arn:aws:es:us-east-1:123456789012:domain/my-domain/*\"}]}"
}
```

Para obtener información detallada, consulte la Referencia de API de Amazon OpenSearch Service.

Migración de índices al almacenamiento UltraWarm

Si termina de escribir en un índice y ya no necesita el rendimiento de búsqueda más rápido posible, realice la migración de caliente a UltraWarm:

```
POST _ultrawarm/migration/my-index/_warm
```

A continuación, verifique el estado de la migración:

```
GET _ultrawarm/migration/my-index/_status

{
  "migration_status": {
    "index": "my-index",
    "state": "RUNNING_SHARD_RELOCATION",
    "migration_type": "HOT_TO_WARM",
    "shard_level_status": {
      "running": 0,
      "total": 5,
      "pending": 3,
      "failed": 0,
      "succeeded": 2
    }
  }
}
```

El estado del índice debe ser verde para realizar una migración. Si migra varios índices en una sucesión rápida, puede obtener un resumen de todas las migraciones en texto sin formato, similar a la API `_cat`:

```
GET _ultrawarm/migration/_status?v

index    migration_type state
my-index HOT_TO_WARM    RUNNING_SHARD_RELOCATION
```

OpenSearch Service migra un índice por vez a UltraWarm. Se pueden tener hasta 200 migraciones en la cola. Cualquier solicitud que supere el límite será rechazada. Para comprobar el número actual de migraciones en la cola, supervise la métrica de `HotToWarmMigrationQueueSize`. Los índices permanecen disponibles durante todo el proceso de migración, sin tiempo de inactividad.

El proceso de migración tiene los siguientes estados:

```
PENDING_INCREMENTAL_SNAPSHOT
RUNNING_INCREMENTAL_SNAPSHOT
FAILED_INCREMENTAL_SNAPSHOT
PENDING_FORCE_MERGE
RUNNING_FORCE_MERGE
FAILED_FORCE_MERGE
PENDING_FULL_SNAPSHOT
RUNNING_FULL_SNAPSHOT
FAILED_FULL_SNAPSHOT
PENDING_SHARD_RELOCATION
RUNNING_SHARD_RELOCATION
FINISHED_SHARD_RELOCATION
```

Como indican estos estados, se pueden producir errores en las migraciones si se realizan durante instantáneas, reubicaciones de particiones o fusiones forzosas. Los errores durante las instantáneas o las reubicaciones de particiones suelen deberse a errores de nodo o problemas de conectividad de S3. La falta de espacio en el disco suele ser la causa subyacente de los errores en las fusiones forzosas.

Una vez finalizada la migración, la misma solicitud `_status` devuelve un error. Si comprueba el índice en ese momento, verá algunos parámetros de configuración que son exclusivos de los índices templados:

```
GET my-index/_settings

{
  "my-index": {
    "settings": {
      "index": {
        "refresh_interval": "-1",
        "auto_expand_replicas": "false",
        "provided_name": "my-index",
        "creation_date": "1599241458998",
        "unassigned": {
```

```
          "node_left": {
            "delayed_timeout": "5m"
          }
        },
        "number_of_replicas": "1",
        "uuid": "GswyCdR0RSq0SJYmzsIpiw",
        "version": {
          "created": "7070099"
        },
        "routing": {
          "allocation": {
            "require": {
              "box_type": "warm"
            }
          }
        },
        "number_of_shards": "5",
        "merge": {
          "policy": {
            "max_merge_at_once_explicit": "50"
          }
        }
      }
    }
  }
}
```

- `number_of_replicas`, en este caso, es el número de réplicas pasivas, que no consumen espacio en disco.
- `routing.allocation.require.box_type` especifica que el índice debe usar nodos templados en lugar de nodos de datos estándar.
- `merge.policy.max_merge_at_once_explicit` especifica el número de segmentos que se van a fusionar simultáneamente durante la migración.

Los índices del almacenamiento caliente son de solo lectura, a menos que los devuelva al almacenamiento en caliente, lo que hace que UltraWarm sea el más adecuado para datos inmutables, como los registros. Puede consultar los índices y eliminarlos, pero no puede agregar, actualizar ni eliminar documentos individuales. Si lo intenta, podría encontrarse con el siguiente error:

```
{
```

```
  "error" : {
    "root_cause" : [
      {
        "type" : "cluster_block_exception",
        "reason" : "index [indexname] blocked by: [TOO_MANY_REQUESTS/12/disk usage
 exceeded flood-stage watermark, index has read-only-allow-delete block];"
      }
    ],
    "type" : "cluster_block_exception",
    "reason" : "index [indexname] blocked by: [TOO_MANY_REQUESTS/12/disk usage exceeded
 flood-stage watermark, index has read-only-allow-delete block];"
  },
  "status" : 429
}
```

Automatizar migraciones

Se recomienda utilizar the section called “Administración de estados de índice” para automatizar el proceso de migración después de que un índice alcance cierta antigüedad o cumpla otras condiciones. Consulte la política de ejemplo que muestra este flujo de trabajo.

Ajuste de migración

Las migraciones de índice al almacenamiento UltraWarm requieren una combinación de fuerza. Cada índice de OpenSearch se compone de un cierto número de particiones, y cada partición está compuesta por un cierto número de segmentos de Lucene. La operación de fusión forzada purga los documentos marcados para su eliminación y conserva espacio en disco. De forma predeterminada, UltraWarm fusiona los índices en un segmento, excepto los índices kNN, donde se utiliza un valor predeterminado de 20.

Puede cambiar este valor hasta 1000 segmentos utilizando la configuración de `index.ultrawarm.migration.force_merge.max_num_segments`. Los valores más altos aceleran el proceso de migración, pero aumentan la latencia de consulta para el índice activo una vez finalizada la migración. Para cambiar la configuración, realice la siguiente solicitud:

```
PUT my-index/_settings
{
  "index": {
    "ultrawarm": {
      "migration": {
        "force_merge": {
```

```
          "max_num_segments": 1
        }
      }
    }
  }
}
```

Para comprobar cuánto tiempo tarda esta etapa del proceso de migración, monitoree la métrica de `HotToWarmMigrationForceMergeLatency`.

Cancelación de migraciones

UltraWarm gestiona las migraciones secuencialmente, en una cola. Si una migración está en la cola, pero aún no se ha iniciado, puede eliminarla de la cola mediante la siguiente solicitud:

```
POST _ultrawarm/migration/_cancel/my-index
```

Si el dominio utiliza un control de acceso detallado, debe tener el permiso de `indices:admin/ultrawarm/migration/cancel` para realizar esta solicitud.

Listado de índices calientes y templados

UltraWarm agrega dos opciones adicionales, similares a `_all`, para facilitar la administración de los índices calientes y templados. Para obtener una lista de todos los índices calientes o templados, realice las siguientes solicitudes:

```
GET _warm
GET _hot
```

Puede utilizar estas opciones en otras solicitudes que especifican índices, por ejemplo:

```
_cat/indices/_warm
_cluster/state/_all/_hot
```

Devolución de índices templados al almacenamiento caliente

Si necesita volver a escribir en un índice, vuelva a migrarlo al almacenamiento en caliente:

```
POST _ultrawarm/migration/my-index/_hot
```

Puede tener hasta 10 migraciones en cola del almacenamiento templado al caliente a la vez. OpenSearch Service procesa las solicitudes de migración de una en una, en el orden en que se colocaron en cola. Para comprobar el número actual, monitoree la métrica de `WarmToHotMigrationQueueSize`.

Una vez finalizada la migración, compruebe la configuración del índice para asegurarse de que satisfaga sus necesidades. Los índices vuelven al almacenamiento caliente con una réplica.

Restauración de índices templados a partir de instantáneas

Además del repositorio estándar para instantáneas automatizadas, UltraWarm agrega un segundo repositorio para índices templados, `cs-ultrawarm`. Cada instantánea de este repositorio contiene solo un índice. Si elimina un índice caliente, su instantánea permanece en el repositorio de `cs-ultrawarm` durante 14 días, al igual que cualquier otra instantánea automatizada.

Cuando restaura una instantánea desde `cs-ultrawarm`, se restaura en el almacenamiento "warm", no en el almacenamiento en caliente. Las instantáneas de los repositorios `cs-automated` y `cs-automated-enc` restauran el almacenamiento en caliente.

Para restaurar una instantánea UltraWarm al almacenamiento "warm"

1. Identifique la instantánea más reciente que contiene el índice que desea restaurar:

```
GET _snapshot/cs-ultrawarm/_all?verbose=false

{
  "snapshots": [{
    "snapshot": "snapshot-name",
    "version": "1.0",
    "indices": [
      "my-index"
    ]
  }]
}
```

> **Note**
>
> De forma predeterminada, la `GET _snapshot/<repo>` operación muestra información detallada sobre los datos, como la hora de inicio, la hora de finalización y la duración de cada instantánea de un repositorio. La `GET _snapshot/<repo>` operación recupera

información de los archivos de cada instantánea contenidos en un repositorio. Si no necesita la hora de inicio, la hora de finalización y la duración y solo necesita el nombre y la información de índice de una instantánea, le recomendamos que utilice el `verbose=false` parámetro al enumerar las instantáneas para minimizar el tiempo de procesamiento y evitar que se agote el tiempo de espera.

2. Si el índice ya existe, elimínelo:

```
DELETE my-index
```

Si no quiere eliminar el índice, debe devolverlo al almacenamiento en caliente y reindexarlo.

3. Restaurar la instantánea:

```
POST _snapshot/cs-ultrawarm/snapshot-name/_restore
```

UltraWarm ignora cualquier configuración de índice que especifique en esta solicitud de restauración, pero puede especificar opciones como `rename_pattern` y `rename_replacement`. Para obtener un resumen de las opciones de restauración de instantáneas de OpenSearch, consulte la documentación de OpenSearch.

Instantáneas manuales de índices templados

Usted puede tomar instantáneas manuales de índices templados, pero no lo recomendamos. El repositorio de automatización `cs-ultrawarm` ya contiene una instantánea para cada índice activo, tomada durante la migración, sin cargo adicional.

De manera predeterminada, OpenSearch Service no incluye los índices templados en las instantáneas manuales. Por ejemplo, la siguiente llamada solo incluye índices calientes:

```
PUT _snapshot/my-repository/my-snapshot
```

Si elige tomar instantáneas manuales de índices templados, se aplican varias consideraciones importantes.

- No puede mezclar índices calientes y templados. Por ejemplo, la siguiente solicitud devuelve un error:

```
PUT _snapshot/my-repository/my-snapshot
{
  "indices": "warm-index-1,hot-index-1",
  "include_global_state": false
}
```

Si incluyen una mezcla de índices calientes y cálidos, comodín (*) también devuelven un error.

- Solo se puede incluir un índice templado por instantánea. Por ejemplo, la siguiente solicitud devuelve un error:

```
PUT _snapshot/my-repository/my-snapshot
{
  "indices": "warm-index-1,warm-index-2,other-warm-indices-*",
  "include_global_state": false
}
```

Esta solicitud se realiza correctamente:

```
PUT _snapshot/my-repository/my-snapshot
{
  "indices": "warm-index-1",
  "include_global_state": false
}
```

- Las instantáneas manuales siempre se restauran al almacenamiento en caliente, incluso si originalmente incluían un índice templado.

Migración de índices templados al almacenamiento frío

Si tiene datos en UltraWarm que consulta con poca frecuencia, considere migrarlos al almacenamiento frío. El almacenamiento frío está diseñado para los datos a los que solo se accede ocasionalmente o que ya no están en uso activo. No puede leer ni escribir en índices fríos, pero puede migrarlos de nuevo al almacenamiento templado sin costo cada vez que necesite consultarlos. Para obtener instrucciones, consulte Migración de índices a cámaras frigoríficas.

Prácticas recomendadas para índices KNN

- El nivel Ultrawarm/Cold está disponible para todos los tipos de motores de indexación KNN. Lo recomendamos para los índices KNN que utilizan el motor Lucene y la búsqueda vectorial optimizada para discos, ya que no es necesario cargar completamente los datos del gráfico en una memoria externa. Al usarlo con motores integrados en memoria nativos, como FAISS y NMSLIB, debe tener en cuenta el tamaño del gráfico de las particiones en los que se buscará activamente y aprovisionar las instancias UltraWarm, preferiblemente del tipo de instancia `uw.large`, en consecuencia. Por ejemplo, si los clientes tienen 2 instancias `uw.large` configuradas, cada una tendrá aproximadamente GiB de `knn.memory.circuit_breaker.limit * 61` de memoria fuera del montón disponibles. Se obtiene un rendimiento óptimo si todas las consultas en caliente se dirigen a particiones cuyo tamaño de gráfico acumulado no supere la memoria externa disponible. La latencia se ve afectada si la memoria disponible es inferior a la necesaria para cargar el gráfico debido a las expulsiones y a la espera de que se disponga de memoria acumulada. Por eso no recomendamos el uso de instancias de `uw.medium` en los casos de uso en los que se utilicen motores integrados en memoria o para casos de mayor rendimiento de búsqueda, independientemente de los motores.
- Los índices KNN que migren a UltraWarm no se fusionarán forzosamente en un solo segmento. Esto evita cualquier impacto en los nodos en caliente y cálidos que podrían enfrentar problemas de falta de memoria (OOM) debido a que el tamaño del grafo se vuelve demasiado grande para los motores en memoria. Debido al aumento del número de segmentos por partición, esto podría consumir más espacio en la memoria caché local y permitir que menos índices migren a la capa cálida. Puede optar por forzar la fusión de índices en un solo segmento utilizando la configuración existente y anulándola antes de migrar los índices a la capa cálida. Para obtener más información, consulte the section called “Ajuste de migración”.
- Si tiene un caso de uso en el que los índices se buscan con poca frecuencia y no satisfacen una carga de trabajo sensible a la latencia, puede optar por migrar esos índices al nivel UltraWarm. Esto lo ayudará a reducir verticalmente la escala de las instancias informáticas de nivel activo y permitir que el cómputo de nivel UltraWarm gestione la consulta en índices de prioridad tan baja. Esto también puede ayudar a aislar los recursos consumidos entre las consultas de los índices de alta y baja prioridad para que no se afecten entre sí.

Deshabilitar UltraWarm

La consola es la forma más sencilla de deshabilitar UltraWarm. Seleccione el dominio, Acciones y Editar la configuración del clúster. Anule la selección de Habilitar nodos de datos cálidos y elija

Guardar los cambios. También puede usar la opción `WarmEnabled` en la API de configuración y en la AWS CLI.

Antes de desactivar UltraWarm, debe eliminar todos los índices templados o migrarlos de nuevo al almacenamiento caliente. Una vez que el almacenamiento templado esté vacío, espere cinco minutos antes de intentar deshabilitar UltraWarm.

Almacenamiento en frío para Amazon OpenSearch Service

El almacenamiento en frío permite almacenar cualquier cantidad de datos históricos o datos a los que se accede con poca frecuencia en el dominio de Amazon OpenSearch Service y analizarlos bajo demanda, a un costo menor que en otros niveles de almacenamiento. El almacenamiento en frío es apropiado si necesita realizar investigaciones periódicas o análisis forenses sobre datos antiguos. Algunos ejemplos prácticos de datos apropiados para el almacenamiento en frío son los registros a los que se accede con poca frecuencia, los datos que se deben conservar para cumplir con los requisitos de conformidad o los registros que tienen valor histórico.

De la misma forma que el almacenamiento UltraWarm, el almacenamiento en frío está respaldado por Amazon S3. Cuando necesite consultar datos almacenados en frío, puede adjuntarlos de forma selectiva a los nodos UltraWarm existentes. Puede administrar la migración y el ciclo de vida de los datos almacenados en frío manualmente o mediante políticas de administración de estados de índice.

Temas

- Requisitos previos
- Requisitos de almacenamiento en frío y consideraciones de rendimiento
- Precio del almacenamiento en frío
- Habilitación del almacenamiento en frío
- Administración de índices fríos en OpenSearch Dashboards
- Migración de índices al almacenamiento frío
- Automatización de las migraciones al almacenamiento en frío
- Cancelación de migraciones al almacenamiento en frío
- Listado de índices almacenados en frío
- Migración de índices fríos al almacenamiento templado

- Restauración de índices fríos a partir de instantáneas
- Cancelación de migraciones del almacenamiento en frío al templado
- Actualización de metadatos de índices almacenados en frío
- Eliminación de índices almacenados en frío
- Deshabilitar el almacenamiento en frío

Requisitos previos

El almacenamiento en frío tiene los siguientes requisitos previos:

- El almacenamiento en frío requiere la versión 7.9 o posterior de OpenSearch o Elasticsearch.
- Para habilitar el almacenamiento en frío en un dominio de OpenSearch Service, también debe habilitar el almacenamiento cálido en el mismo dominio.
- Para utilizar el almacenamiento en frío, los dominios deben tener nodos maestros dedicados.
- Si un dominio utiliza un tipo de instancia T2 o T3 para los nodos de datos, no puede utilizar el almacenamiento en frío.
- Si su índice utiliza k-NN aproximado (`"index.knn":true`), puede trasladarlo al almacenamiento en frío a partir de la versión 2.17 en adelante. Los dominios de versiones anteriores a la 2.17 pueden actualizarse a la 2.17 para utilizar esta funcionalidad, pero los índices KNN creados en versiones anteriores a la 2.x no pueden migrar a frío.
- Si el dominio utiliza el control de acceso detallado, los usuarios no administradores deben estar asignados al rol `cold_manager` en OpenSearch Dashboards para poder administrar índices almacenados en frío.

> **Note**
>
> El rol `cold_manager` puede no existir en algunos dominios de OpenSearch Service preexistentes. Si no ve el rol en el panel, debe crearlo de forma manual.

Configuración de permisos

Si habilita el almacenamiento en frío en un dominio de OpenSearch Service preexistente, el rol `cold_manager` podría no estar definido en el dominio. Si el dominio utiliza el control de acceso

detallado, los usuarios que no sean administradores deben estar asignados a este rol para poder administrar índices almacenados en frío. Para crear el rol `cold_manager` de forma manual, siga estos pasos:

1. En OpenSearch Dashboards, vaya a Seguridad y elija Permisos.
2. Seleccione Crear grupo de acciones y configure los siguientes grupos:

Nombre del grupo	Permisos
`cold_cluster`	• `cluster:monitor/nodes/stats` • `cluster:admin/ultrawarm*` • `cluster:admin/cold/*`
`cold_index`	• `indices:monitor/stats` • `indices:data/read/minmax` • `indices:admin/ultrawarm/migration/get` • `indices:admin/ultrawarm/migration/cancel`

3. Seleccione Roles y, a continuación, Crear rol.
4. Nombre el rol cold_manager.
5. En Permisos de clúster, seleccione el grupo `cold_cluster` que creó.
6. En Índice, ingrese `*`.
7. En Permisos de índice, seleccione el grupo `cold_index` que creó.
8. Seleccione Crear.
9. Después de crear el rol, asígnelo a cualquier rol de usuario o backend que administre índices almacenados en frío.

Requisitos de almacenamiento en frío y consideraciones de rendimiento

Dado que el almacenamiento frío utiliza Amazon S3, no conlleva ninguno de los gastos del almacenamiento caliente, tales como réplicas, espacio reservado de Linux y espacio reservado de OpenSearch Service. El almacenamiento en frío no tiene tipos de instancias específicos porque no tiene ninguna capacidad informática asociada a él. Puede almacenar cualquier cantidad de datos en el almacenamiento en frío. Monitoree la métrica `ColdStorageSpaceUtilization` en Amazon CloudWatch para ver cuánto espacio de almacenamiento en frío utiliza.

Precio del almacenamiento en frío

Como sucede con el almacenamiento UltraWarm, con el almacenamiento en frío solo paga por el almacenamiento de datos. No hay ningún costo informático para los datos en frío y no incurrirá en gastos si no hay datos en el almacenamiento en frío.

No incurrirá en cargos de transferencia al mover datos entre los almacenamientos en frío y templado. Mientras se realiza la migración de los índices entre el almacenamiento templado y el frío, sigue pagando por una sola copia del índice. Una vez finalizada la migración, el índice se factura de acuerdo con el nivel de almacenamiento al que se migró. Para obtener más información sobre los precios del almacenamiento frío, consulte Precios de Amazon OpenSearch Service.

Habilitación del almacenamiento en frío

La consola es la forma más sencilla de crear un dominio que utiliza el almacenamiento en frío. Mientras crea el dominio, primero elija Habilitar nodos de datos en caliente, porque debe habilitar el almacenamiento en caliente en el mismo dominio. A continuación, elija Habilitar almacenamiento en frío.

El mismo proceso funciona en dominios existentes, siempre que cumplan los requisitos previos. Incluso después de que el estado del dominio cambie de En proceso a Activo, es posible que el almacenamiento en frío no esté disponible para su utilización por varias horas.

También puede utilizar AWS CLI o la API de configuración para habilitar el almacenamiento en frío.

Ejemplo de comando de la CLI

El comando AWS CLI crea un dominio con tres nodos de datos, tres nodos maestros dedicados, almacenamiento en frío habilitado y control de acceso detallado habilitado:

```
aws opensearch create-domain \
  --domain-name my-domain \
  --engine-version Opensearch_1.0 \
  --cluster-
config ColdStorageOptions={Enabled=true},WarmEnabled=true,WarmCount=4,WarmType=ultrawarm1.mediu
 \
  --ebs-options EBSEnabled=true,VolumeType=gp2,VolumeSize=11 \
  --node-to-node-encryption-options Enabled=true \
  --encryption-at-rest-options Enabled=true \
  --domain-endpoint-options EnforceHTTPS=true,TLSSecurityPolicy=Policy-Min-
TLS-1-2-2019-07 \
```

```
  --advanced-security-options
 Enabled=true,InternalUserDatabaseEnabled=true,MasterUserOptions='{MasterUserName=master-
user,MasterUserPassword=master-password}' \
  --region us-east-2
```

Para obtener más información, consulte la Referencia de comandos de la AWS CLI.

Ejemplo de solicitud a la API de configuración

La siguiente solicitud a la API de configuración crea un dominio con tres nodos de datos, tres nodos maestros dedicados, almacenamiento en frío habilitado y control de acceso detallado habilitado:

```
POST https://es.us-east-2.amazonaws.com/2021-01-01/opensearch/domain
{
  "ClusterConfig": {
    "InstanceCount": 3,
    "InstanceType": "r6g.large.search",
    "DedicatedMasterEnabled": true,
    "DedicatedMasterType": "r6g.large.search",
    "DedicatedMasterCount": 3,
    "ZoneAwarenessEnabled": true,
    "ZoneAwarenessConfig": {
      "AvailabilityZoneCount": 3
     },
    "WarmEnabled": true,
    "WarmCount": 4,
    "WarmType": "ultrawarm1.medium.search",
    "ColdStorageOptions": {
       "Enabled": true
     }
  },
  "EBSOptions": {
    "EBSEnabled": true,
    "VolumeType": "gp2",
    "VolumeSize": 11
  },
  "EncryptionAtRestOptions": {
    "Enabled": true
  },
  "NodeToNodeEncryptionOptions": {
    "Enabled": true
  },
  "DomainEndpointOptions": {
```

```
    "EnforceHTTPS": true,
    "TLSSecurityPolicy": "Policy-Min-TLS-1-2-2019-07"
  },
   "AdvancedSecurityOptions": {
    "Enabled": true,
    "InternalUserDatabaseEnabled": true,
    "MasterUserOptions": {
      "MasterUserName": "master-user",
      "MasterUserPassword": "master-password"
    }
  },
  "EngineVersion": "Opensearch_1.0",
  "DomainName": "my-domain"
}
```

Para obtener información detallada, consulte la Referencia de API de Amazon OpenSearch Service.

Administración de índices fríos en OpenSearch Dashboards

Puede administrar índices calientes, templados y fríos con la interfaz de Dashboards existente en el dominio de OpenSearch Service. Dashboards permite migrar índices entre el almacenamiento frío y el templado, así como supervisar el estado de la migración de los índices, sin utilizar la CLI ni la API de configuración. Para obtener más información, consulte Administración de índices en OpenSearch Dashboards.

Migración de índices al almacenamiento frío

Cuando se migran índices al almacenamiento frío, se debe proporcionar un intervalo de tiempo a los datos para facilitar el descubrimiento. Puede elegir un campo de marca temporal basado en los datos del índice, proporcionar de forma manual una marca temporal inicial y final o elegir no especificar una.

Parámetro	Valor admitido	Descripción
`timestamp_field`	Campo de fecha/hora del mapeo de índices.	Los valores mínimo y máximo del campo proporcionado se calculan y almacenan como metadatos `start_time` y `end_time` para el índice almacenado en frío.

Parámetro	Valor admitido	Descripción
`start_time` y `end_time`	Utilice uno de los siguientes formatos: • strict_date_optional_time. Por ejemplo, `yyyy-MM-dd'T'HH:mm:ss.SSSZ` o `yyyy-MM-dd` • Fecha de inicio en milisegundos	Los valores proporcionados se almacenan como metadatos `start_time` y `end_time` para el índice almacenado en frío.

Si no desea especificar una marca temporal, agregue `?ignore=timestamp` a la solicitud en su lugar.

La siguiente solicitud migra un índice del almacenamiento templado al almacenamiento en frío y proporciona horas de inicio y finalización para los datos de ese índice:

```
POST _ultrawarm/migration/my-index/_cold
  {
    "start_time": "2020-03-09",
    "end_time": "2020-03-09T23:00:00Z"
  }
```

A continuación, verifique el estado de la migración:

```
GET _ultrawarm/migration/my-index/_status

{
  "migration_status": {
    "index": "my-index",
    "state": "RUNNING_METADATA_RELOCATION",
    "migration_type": "WARM_TO_COLD"
  }
}
```

OpenSearch Service migra un índice por vez al almacenamiento en frío. Se pueden tener hasta 100 migraciones en la cola. Cualquier solicitud que supere el límite será rechazada.

Para comprobar el número actual de migraciones en la cola, supervise la métrica de `WarmToColdMigrationQueueSize`. El proceso de migración tiene los siguientes estados:

```
ACCEPTED_COLD_MIGRATION - Migration request is accepted and queued.
RUNNING_METADATA_MIGRATION - The migration request was selected for execution and
 metadata is migrating to cold storage.
FAILED_METADATA_MIGRATION - The attempt to add index metadata has failed and all
 retries are exhausted.
PENDING_INDEX_DETACH - Index metadata migration to cold storage is completed. Preparing
 to detach the warm index state from the local cluster.
RUNNING_INDEX_DETACH - Local warm index state from the cluster is being removed. Upon
 success, the migration request will be completed.
FAILED_INDEX_DETACH - The index detach process failed and all retries are exhausted.
```

Automatización de las migraciones al almacenamiento en frío

Se recomienda utilizar la Administración de estados de índice para automatizar el proceso de migración después de que un índice alcance cierta antigüedad o cumpla otras condiciones. Consulte el ejemplo de política, que muestra cómo migrar automáticamente los índices del almacenamiento caliente al almacenamiento UltraWarm y luego al almacenamiento frío.

> **Note**
>
> Es necesario un `timestamp_field` explícito para mover los índices al almacenamiento frío mediante una política de administración de estados de índice.

Cancelación de migraciones al almacenamiento en frío

Si una migración al almacenamiento en frío está en cola o con un estado de error, puede cancelar la migración mediante la siguiente solicitud:

```
POST _ultrawarm/migration/_cancel/my-index

{
  "acknowledged" : true
}
```

Si el dominio utiliza un control de acceso detallado, necesita el permiso `indices:admin/ultrawarm/migration/cancel` para realizar esta solicitud.

Listado de índices almacenados en frío

Antes de realizar consultas, puede ver una lista de los índices del almacenamiento frío para decidir cuáles se deben migrar a al almacenamiento UltraWarm con el fin de realizar más análisis. La siguiente solicitud enumera todos los índices almacenados en frío, ordenados por nombre de índice:

```
GET _cold/indices/_search
```

Respuesta de ejemplo


```
{
  "pagination_id" : "je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY",
  "total_results" : 3,
  "indices" : [
    {
      "index" : "my-index-1",
      "index_cold_uuid" : "hjEoh26mRRCFxRIMdgvLmg",
      "size" : 10339,
      "creation_date" : "2021-06-28T20:23:31.206Z",
      "start_time" : "2020-03-09T00:00Z",
      "end_time" : "2020-03-09T23:00Z"
    },
    {
      "index" : "my-index-2",
      "index_cold_uuid" : "0vIS2n-oROmOWDFmwFIgdw",
      "size" : 6068,
      "creation_date" : "2021-07-15T19:41:18.046Z",
      "start_time" : "2020-03-09T00:00Z",
      "end_time" : "2020-03-09T23:00Z"
    },
    {
      "index" : "my-index-3",
      "index_cold_uuid" : "EaeXOBodTLiDYcivKsXVLQ",
      "size" : 32403,
      "creation_date" : "2021-07-08T00:12:01.523Z",
      "start_time" : "2020-03-09T00:00Z",
      "end_time" : "2020-03-09T23:00Z"
    }
  ]
}
```

Filtrado

Puede filtrar los índices fríos según un patrón de índice basado en prefijos y variaciones del intervalo de tiempo.

La siguiente solicitud presenta una lista de los índices que coinciden con el patrón de prefijo `event-*`:

```
GET _cold/indices/_search
 {
   "filters":{
      "index_pattern": "event-*"
   }
 }
```

Respuesta de ejemplo

```
{
  "pagination_id" : "je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY",
  "total_results" : 1,
  "indices" : [
    {
      "index" : "events-index",
      "index_cold_uuid" : "4eFiab7rRfSvp3slrIsIKA",
      "size" : 32263273,
      "creation_date" : "2021-08-18T18:25:31.845Z",
      "start_time" : "2020-03-09T00:00Z",
      "end_time" : "2020-03-09T23:00Z"
    }
  ]
}
```

La siguiente solicitud devuelve índices cuyos campos de metadatos `start_time` y `end_time` están comprendidos entre `2019-03-01` y `2020-03-01`:

```
GET _cold/indices/_search
{
  "filters": {
    "time_range": {
      "start_time": "2019-03-01",
      "end_time": "2020-03-01"
```

```
    }
  }
}
```

Respuesta de ejemplo

```
{
  "pagination_id" : "je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY",
  "total_results" : 1,
  "indices" : [
    {
      "index" : "my-index",
      "index_cold_uuid" : "4eFiab7rRfSvp3slrIsIKA",
      "size" : 32263273,
      "creation_date" : "2021-08-18T18:25:31.845Z",
      "start_time" : "2019-05-09T00:00Z",
      "end_time" : "2019-09-09T23:00Z"
    }
  ]
}
```

Ordenar

Puede ordenar los índices fríos por campos de metadatos, como el nombre del índice o su tamaño. La siguiente solicitud presenta una lista de todos los índices ordenados por tamaño en orden descendente:

```
GET _cold/indices/_search
 {
 "sort_key": "size:desc"
 }
```

Respuesta de ejemplo

```
{
  "pagination_id" : "je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY",
  "total_results" : 5,
  "indices" : [
    {
      "index" : "my-index-6",
      "index_cold_uuid" : "4eFiab7rRfSvp3slrIsIKA",
```

```
      "size" : 32263273,
      "creation_date" : "2021-08-18T18:25:31.845Z",
      "start_time" : "2020-03-09T00:00Z",
      "end_time" : "2020-03-09T23:00Z"
    },
    {
      "index" : "my-index-9",
      "index_cold_uuid" : "mbD3ZRVDRI6ONqgEOsJyUA",
      "size" : 57922,
      "creation_date" : "2021-07-07T23:41:35.640Z",
      "start_time" : "2020-03-09T00:00Z",
      "end_time" : "2020-03-09T23:00Z"
    },
    {
      "index" : "my-index-5",
      "index_cold_uuid" : "EaeXOBodTLiDYcivKsXVLQ",
      "size" : 32403,
      "creation_date" : "2021-07-08T00:12:01.523Z",
      "start_time" : "2020-03-09T00:00Z",
      "end_time" : "2020-03-09T23:00Z"
    }
  ]
}
```

Otras claves de ordenamiento válidas son `start_time:asc/desc`, `end_time:asc/desc` y `index_name:asc/desc`.

Paginación

Puede paginar una lista de índices almacenados en frío. Configure el número de índices que se deben devolver por página con el parámetro `page_size` (el valor predeterminado es 10). Cada solicitud `_search` aplicada a los índices fríos devuelve un `pagination_id` que se puede utilizar para llamadas posteriores.

La siguiente solicitud realiza la paginación de los resultados de una solicitud `_search` de los índices fríos y muestra los siguientes 100 resultados:

```
GET _cold/indices/_search?page_size=100
{
"pagination_id": "je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY"
}
```

Migración de índices fríos al almacenamiento templado

Después de reducir la lista de índices fríos con los criterios de filtrado de la sección anterior, vuelva a migrarlos a UltraWarm, donde puede consultar los datos y utilizarlos para crear visualizaciones.

La siguiente solicitud vuelve a migrar dos índices fríos al almacenamiento templado:

```
POST _cold/migration/_warm
 {
 "indices": "my-index1,my-index2"
 }


{
  "acknowledged" : true
}
```

Para verificar el estado de la migración y recuperar el ID de migración, envíe la siguiente solicitud:

```
GET _cold/migration/_status
```

Respuesta de ejemplo

```
{
  "cold_to_warm_migration_status" : [
    {
      "migration_id" : "tyLjXCA-S76zPQbPVHkOKA",
      "indices" : [
        "my-index1,my-index2"
      ],
      "state" : "RUNNING_INDEX_CREATION"
    }
  ]
}
```

Para obtener información de migración específica del índice, incluya el nombre del índice:

```
GET _cold/migration/my-index/_status
```

En lugar de especificar un índice, puede ver una lista de los índices por su estado de migración actual. Los valores válidos son `_failed`, `_accepted` y `_all`.

El siguiente comando obtiene el estado de todos los índices de una única solicitud de migración:

```
GET _cold/migration/_status?migration_id=my-migration-id
```

Recupere el ID de migración mediante la solicitud de estado. Para obtener información detallada sobre la migración, agregue `&verbose=true`.

Puede migrar índices del almacenamiento de frío al almacenamiento templado en lotes de 10 o menos, con un máximo de 100 solicitudes que se migran en forma simultánea. Cualquier solicitud que supere el límite será rechazada. Para comprobar el número actual de migraciones que se están llevando a cabo actualmente, supervise la métrica de `ColdToWarmMigrationQueueSize`. El proceso de migración tiene los siguientes estados:

```
ACCEPTED_MIGRATION_REQUEST - Migration request is accepted and queued.
RUNNING_INDEX_CREATION - Migration request is picked up for processing and will create
 warm indexes in the cluster.
PENDING_COLD_METADATA_CLEANUP - Warm index is created and the migration service will
 attempt to clean up cold metadata.
RUNNING_COLD_METADATA_CLEANUP - Cleaning up cold metadata from the indexes migrated to
 warm storage.
FAILED_COLD_METADATA_CLEANUP - Failed to clean up metadata in the cold tier.
FAILED_INDEX_CREATION - Failed to create an index in the warm tier.
```

Restauración de índices fríos a partir de instantáneas

Si necesita restaurar un índice en frío eliminado, puede restaurarlo de nuevo en el nivel cálido siguiendo las instrucciones que aparecen en el documento the section called “Restauración de índices templados a partir de instantáneas” y, a continuación, migrando de nuevo el índice al nivel frío. No puede restaurar un índice en frío eliminado directamente al nivel frío. OpenSearch Service conserva los índices fríos durante 14 días después de que se hayan eliminado.

Cancelación de migraciones del almacenamiento en frío al templado

Si se pone en cola una migración de índice del almacenamiento en frío al templado o si presenta un estado de error, puede cancelarla mediante la siguiente solicitud:

```
POST _cold/migration/my-index/_cancel

{
```

```
  "acknowledged" : true
}
```

Para cancelar la migración de un lote de índices (máximo 10 a la vez), especifique el ID de migración:

```
POST _cold/migration/_cancel?migration_id=my-migration-id

{
  "acknowledged" : true
}
```

Recupere el ID de migración mediante la solicitud de estado.

Actualización de metadatos de índices almacenados en frío

Puede actualizar los campos `start_time` y `end_time` de un índice almacenado en frío:

```
PATCH _cold/my-index
 {
 "start_time": "2020-01-01",
 "end_time": "2020-02-01"
 }
```

No puede actualizar el `timestamp_field` de un índice de almacenamiento en frío.

> **Note**
>
> OpenSearch Dashboards no admite el método PATCH. Utilice curl, Postman o algún otro método para actualizar metadatos almacenados en frío.

Eliminación de índices almacenados en frío

Si no utiliza una política de ISM, puede eliminar los índices fríos de forma manual. La siguiente solicitud elimina un índice almacenado en frío:

```
DELETE _cold/my-index
```

```
{
  "acknowledged" : true
}
```

Deshabilitar el almacenamiento en frío

La consola de OpenSearch Service es la forma más sencilla de deshabilitar el almacenamiento en frío. Seleccione el dominio y elija Acciones, Editar la configuración del clúster, luego anule la selección de Habilitar el almacenamiento en frío).

Para utilizar la AWSCLI o la API de configuración, en `ColdStorageOptions`, establezca `"Enabled"="false"`.

Antes de desactivar el almacenamiento frío, debe eliminar todos los índices fríos o volver a migrarlos al almacenamiento templado; de lo contrario, la acción de desactivación no funcionará.

Instancias optimizadas para OpenSearch (OR1) para dominios de Amazon OpenSearch Service

La familia de instancias optimizadas para OpenSearch de Amazon OpenSearch Service es una solución rentable para almacenar grandes volúmenes de datos. Un dominio con instancias optimizadas para OpenSearch (OR1, OR2, OM2) utiliza volúmenes `gp3` o `io1` de Amazon Elastic Block Store (Amazon EBS) como almacenamiento principal y los datos se copian de manera sincrónica en Amazon S3 a medida que llegan. Esta estructura de almacenamiento proporciona un mayor rendimiento de indexación con una alta durabilidad. La familia de instancias optimizadas para OpenSearch también admite la recuperación automática de datos en caso de error. Para obtener más información acerca de las opciones del tipo de instancias optimizadas para OpenSearch, consulte the section called “Tipos de instancias de generación actual”.

Si está indexando cargas de trabajo de análisis operativo pesadas, como análisis de registros, observabilidad o análisis de seguridad, puede beneficiarse de la mejora del rendimiento y la eficiencia de computación de las instancias optimizadas para OpenSearch. Además, la recuperación automática de datos que ofrecen las instancias optimizadas para OpenSearch mejora la fiabilidad general del dominio.

El Servicio OpenSearch envía a Amazon CloudWatch métricas relacionadas con el almacenamiento de instancias optimizadas para OpenSearch. Para ver una lista de las métricas disponibles, consulte ???.

Las instancias optimizadas para OpenSearch están disponibles bajo demanda o con precios de instancias reservadas, con una tarifa por hora para las instancias y el almacenamiento aprovisionado en Amazon EBS y Amazon S3.

Temas

- Limitaciones
- Ajustes para mejorar el rendimiento de la ingesta
- En qué se diferencian las instancias optimizadas para OpenSearch de otras instancias
- En qué se diferencian las instancias optimizadas para OpenSearch de las instancias UltraWarm
- Aprovisionamiento de un dominio con instancias optimizadas de OpenSearch

Limitaciones

Tenga en cuenta las siguientes limitaciones cuando utilice instancias optimizadas para OpenSearch para su dominio.

- Los dominios recién creados deben ejecutarse en la versión 2.11 o una posterior de OpenSearch.
- Los dominios existentes deben ejecutarse en la versión 2.15 o una posterior de OpenSearch.
- Su dominio debe tener activado el cifrado en reposo. Para obtener más información, consulte ???.
- Si el dominio usa nodos maestros dedicados, deben usar instancias de Graviton. Para obtener más información sobre nodos maestros dedicados, consulte ???.
- El intervalo de actualización de los índices de las instancias optimizadas para OpenSearch debe ser de 10 segundos o más. El intervalo de actualización predeterminado para las instancias optimizadas para OpenSearch es de 10 segundos.

Ajustes para mejorar el rendimiento de la ingesta

Para obtener el mejor rendimiento de indexación de las instancias optimizadas para OpenSearch, recomendamos hacer lo siguiente:

- Use tamaños de carga masivos para mejorar la utilización del búfer. El tamaño recomendado es 10 MB.
- Utilice varios clientes para mejorar el rendimiento del procesamiento paralelo.
- Establezca la cantidad de particiones principales activas para que coincida con la cantidad de nodos de datos para maximizar el uso de los recursos.

En qué se diferencian las instancias optimizadas para OpenSearch de otras instancias

Las instancias optimizadas para OpenSearch se diferencian de las instancias no optimizadas en los siguientes aspectos:

- En el caso de las instancias optimizadas para OpenSearch, la indexación solo se lleva a cabo en las particiones principales.
- Si las instancias optimizadas de OpenSearch están configuradas con réplicas, la tasa de indexación puede parecer inferior a la real. Por ejemplo, si hay una partición primario y una partición réplica, la velocidad de indexación podría mostrar una tasa de 1000 cuando la velocidad de indexación real es 2000.
- Las instancias optimizadas de OpenSearch llevan a cabo operaciones de búfer antes de enviarlas a un origen remoto. Esto se traduce en una latencia de ingesta superior.

> **Note**
>
> La métrica `IndexingLatency` no se ve afectada, ya que no incluye el tiempo de sincronización de translog.

- Las particiones de réplica pueden estar unos segundos por detrás de las particiones principales. Usted puede supervisar el retraso usando la métrica `ReplicationLagMaxTime` de Amazon CloudWatch.

En qué se diferencian las instancias optimizadas para OpenSearch de las instancias UltraWarm

OpenSearch Service proporciona instancias de UltraWarm que son una forma rentable de almacenar grandes cantidades de datos de solo lectura. Tanto las instancias optimizadas para OpenSearch como las instancias UltraWarm almacenan datos localmente en Amazon EBS y de forma remota en Amazon S3. Sin embargo, las instancias optimizadas para OpenSearch y las instancias UltraWarm difieren en varios aspectos importantes:

- Las instancias optimizadas para OpenSearch guardan una copia de los datos tanto en el almacén local como en el remoto. En instancias de UltraWarm, los datos se guardan principalmente en un

almacén remoto para reducir los costos de almacenamiento. Según sus patrones de uso, los datos se pueden mover al almacenamiento local.

- Las instancias optimizadas para OpenSearch están activas y pueden aceptar operaciones de lectura y escritura, mientras que los datos de las instancias UltraWarm son de solo lectura hasta que se devuelvan manualmente al almacenamiento de acceso frecuente.
- UltraWarm se basa en las instantáneas de índice para la durabilidad de los datos. Las instancias optimizadas para OpenSearch, en comparación, lleva a cabo la replicación y la recuperación entre bastidores. En caso de que aparezca un índice rojo, las instancias optimizadas para OpenSearch restaurarán automáticamente las particiones que falten en el almacenamiento remoto en Amazon S3. El tiempo de recuperación depende del volumen de datos que deban recuperarse.

Para obtener más información acerca del almacenamiento de UltraWarm, consulte ???.

Aprovisionamiento de un dominio con instancias optimizadas de OpenSearch

Usted puede seleccionar instancias optimizadas para OpenSearch para sus nodos de datos cuando crea un dominio nuevo con las Consola de administración de AWS o las AWS Command Line Interface (AWS CLI). A continuación, puede indexar y consultar los datos mediante las operaciones y herramientas existentes.

Consola

1. Vaya a la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/.
2. En el panel de navegación izquierdo, seleccione Dominios.
3. Elija Crear un dominio.
4. En la sección Número de nodos de datos, expanda el menú de la familia de instancias y elija OpenSearch optimizado.
5. Elija el tipo de instancia y otros ajustes de almacenamiento.
6. En la sección Cifrado, asegúrate de que esté seleccionada la opción Habilitar el cifrado de datos en reposo.
7. Configure el resto del dominio y seleccione Crear.

AWS CLI

Para aprovisionar un dominio que utilice el almacenamiento optimizado de OpenSearch mediante AWS CLI, debe proporcionar el valor del tamaño del tipo de instancia específico (como OR1, OR2 u OM2) en el `InstanceType`.

El siguiente ejemplo crea un dominio con instancias OR1 de tamaño `2xlarge` y habilita el cifrado en reposo.

```
aws opensearch create-domain \
  --domain-name test-domain \
  --engine-version OpenSearch_2.11 \
  --cluster-config
 "InstanceType=or1.2xlarge.search,InstanceCount=3,DedicatedMasterEnabled=true,DedicatedMasterTy
 \
  --ebs-options "EBSEnabled=true,VolumeType=gp3,VolumeSize=200" \
  --encryption-at-rest-options Enabled=true \
  --advanced-security-options
 "Enabled=true,InternalUserDatabaseEnabled=true,MasterUserOptions={MasterUserName=test-
user,MasterUserPassword=test-password}" \
  --node-to-node-encryption-options Enabled=true \
  --domain-endpoint-options EnforceHTTPS=true \
  --access-policies '{"Version": "2012-10-17",       "Statement":
[{"Effect":"Allow","Principal":{"AWS":"*"},"Action":"es:*","Resource":"arn:aws:es:us-
east-1:account-id:domain/test-domain/*"}]}'
```

Administración de estados de índice en Amazon OpenSearch Service

La administración de estados de índice (ISM) de Amazon OpenSearch Service permite definir políticas administradas personalizadas para automatizar tareas rutinarias y aplicarlas a índices y patrones de índices. Ya no es necesario configurar y administrar procesos externos para ejecutar las operaciones de índice.

Una política incluye un estado predeterminado y una lista de estados para que el índice pase de un estado al otro. Dentro de cada estado, se puede definir una lista de acciones para realizar y las condiciones que activan estas transiciones. Un caso de uso típico es eliminar periódicamente los índices antiguos después de un determinado periodo de tiempo. Por ejemplo, puede definir una política que mueva el índice a un estado `read_only` después de 30 días y, a continuación, lo elimine después de 90 días.

Después de adjuntar una política a un índice, ISM crea un trabajo que se ejecuta cada 5 a 8 minutos (o cada 30 a 48 minutos en los clústeres de la versión 1.3 y anteriores) para realizar acciones de políticas, verificar condiciones y pasar el índice a estados diferentes. El tiempo base para que este trabajo se ejecute es cada 5 minutos, además de una fluctuación aleatoria del 0 al 60 % para asegurarse de que no se produzca un aumento de la actividad de todos los índices al mismo tiempo. ISM no ejecuta trabajos si el estado del clúster es rojo.

ISM requiere OpenSearch o Elasticsearch 6.8 o posterior.

> **Note**
>
> Esta documentación proporciona una breve información general de ISM y varios ejemplos de políticas. También explica en qué se diferencia ISM para los dominios de Amazon OpenSearch Service de ISM en los clústeres de OpenSearch autogestionados. Para obtener la documentación completa sobre ISM, que incluye una referencia exhaustiva de los parámetros, las descripciones de cada configuración y una referencia de la API, consulte Index State Management en la documentación de OpenSearch.

> **Important**
>
> Ya no se pueden utilizar plantillas de índice para aplicar políticas de ISM a los índices recién creados. Puede seguir administrando automáticamente los índices recién creados con el campo de plantillas de ISM. Esta actualización presenta un cambio interrumpido que afecta a las plantillas existentes de CloudFormation mediante esta configuración.

Crear una política de ISM

Para empezar a utilizar la administración de estados de índice

1. Abra la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. Seleccione el dominio para el que desea crear una política de ISM.
3. En el panel de control del dominio, navegue hasta la URL de OpenSearch Dashboards e inicie sesión con su nombre de usuario maestro y contraseña. La URL tiene este formato:

   ```
   domain-endpoint/_dashboards/
   ```

4. Abra el panel de navegación izquierdo en OpenSearch Dashboards y elija Administración de índices, luego Crear política.
5. Utilice el editor visual o el editor de JSON para crear políticas. Se recomienda utilizar el editor visual, ya que ofrece una forma más estructurada de definir las políticas. Para obtener ayuda para crear políticas, consulte las políticas de muestra a continuación.
6. Después de crear una política, puede adjuntarla a uno o más índices:

```
POST _plugins/_ism/add/my-index
{
  "policy_id": "my-policy-id"
}
```

> **Note**
>
> Si su dominio ejecuta una versión de Elasticsearch heredada, utilice `_opendistro` en lugar de `_plugins`.

Como alternativa, puede elegir el índice en OpenSearch Dashboards y, a continuación, elegir Aplicar política.

Ejemplos de política

Los siguientes ejemplos de política muestran cómo automatizar los casos de uso comunes de ISM.

Traspaso de almacenamiento en caliente a templado y a frío

Este ejemplo de política mueve un índice del almacenamiento en caliente al almacenamiento UltraWarm y, finalmente, al almacenamiento en frío. A continuación, elimina el índice.

El índice se encuentra inicialmente en el estado `hot`. Transcurridos 10 días, ISM lo mueve al estado `warm`. 80 días más tarde, cuando el índice tiene 90 días, ISM lo mueve al estado `cold`. Luego de un año, el servicio envía una notificación a una sala de Amazon Chime que indica que el índice está siendo eliminado y, a continuación, lo elimina permanentemente.

Tenga en cuenta que los índices fríos requieren la operación `cold_delete`, en lugar de la operación `delete` normal. Tenga en cuenta también que es necesario un `timestamp_field` explícito en los datos para administrar índices fríos con ISM.

```
{
  "policy": {
    "description": "Demonstrate a hot-warm-cold-delete workflow.",
    "default_state": "hot",
    "schema_version": 1,
    "states": [{
        "name": "hot",
        "actions": [],
        "transitions": [{
          "state_name": "warm",
          "conditions": {
            "min_index_age": "10d"
          }
        }]
      },
      {
        "name": "warm",
        "actions": [{
          "warm_migration": {},
          "retry": {
            "count": 5,
            "delay": "1h"
          }
        }],
        "transitions": [{
          "state_name": "cold",
          "conditions": {
            "min_index_age": "90d"
          }
        }]
      },
      {
        "name": "cold",
        "actions": [{
            "cold_migration": {
              "timestamp_field": "<your timestamp field>"
            }
          }
        ],
        "transitions": [{
          "state_name": "delete",
          "conditions": {
             "min_index_age": "365d"
```

```
          }
        }]
      },
      {
        "name": "delete",
        "actions": [{
          "notification": {
            "destination": {
              "chime": {
                "url": "<URL>"
              }
            },
            "message_template": {
              "source": "The index {{ctx.index}} is being deleted."
            }
          }
        },
        {
          "cold_delete": {}
        }]
      }
    ]
  }
}
```

Reducir el recuento de réplicas

Este ejemplo de política reduce el recuento de réplicas a cero, después de siete días, para conservar espacio en disco y, a continuación, elimina el índice después de 21 días. Esta política supone que su índice no es crítico y ya no recibe solicitudes de escritura; tener cero réplicas conlleva cierto riesgo de pérdida de datos.

```
{
  "policy": {
    "description": "Changes replica count and deletes.",
    "schema_version": 1,
    "default_state": "current",
    "states": [{
        "name": "current",
        "actions": [],
        "transitions": [{
          "state_name": "old",
          "conditions": {
```

```
            "min_index_age": "7d"
          }
        }]
      },
      {
        "name": "old",
        "actions": [{
          "replica_count": {
            "number_of_replicas": 0
          }
        }],
        "transitions": [{
          "state_name": "delete",
          "conditions": {
            "min_index_age": "21d"
          }
        }]
      },
      {
        "name": "delete",
        "actions": [{
          "delete": {}
        }],
        "transitions": []
      }
    ]
  }
}
```

Tomar una instantánea de índice

Este ejemplo de política utiliza la operación `snapshot` para tomar una instantánea de un índice tan pronto como contenga al menos un documento. `repository` es el nombre del repositorio de instantáneas manual que registró en Amazon S3. `snapshot` es el nombre de la instantánea. Para obtener más información sobre los requisitos previos de una instantánea y los pasos para registrar un repositorio, consulte the section called “Crear instantáneas de índice”.

```
{
  "policy": {
    "description": "Takes an index snapshot.",
    "schema_version": 1,
    "default_state": "empty",
    "states": [{
```

```
        "name": "empty",
        "actions": [],
        "transitions": [{
          "state_name": "occupied",
          "conditions": {
            "min_doc_count": 1
          }
        }]
      },
      {
        "name": "occupied",
        "actions": [{
          "snapshot": {
            "repository": "<my-repository>",
            "snapshot": "<my-snapshot>"
            }
          }],
          "transitions": []
      }
    ]
  }
}
```

Plantillas de ISM

Puede configurar un campo `ism_template` en una política, de modo que cuando se cree un índice que coincida con el patrón de plantilla, la política se adjuntará automáticamente a ese índice. En este ejemplo, cualquier índice que cree con un nombre que comience por “registro” se hace coincidir automáticamente con la política de ISM `my-policy-id`:

```
PUT _plugins/_ism/policies/my-policy-id
{
  "policy": {
    "description": "Example policy.",
    "default_state": "...",
    "states": [...],
    "ism_template": {
      "index_patterns": ["log*"],
      "priority": 100
    }
  }
}
```

Para ver un ejemplo más detallado, consulte Ejemplo de política con plantillas de ISM para reversión automática.

Diferencias

En comparación con OpenSearch y Elasticsearch, ISM para Amazon OpenSearch Service presenta varias diferencias.

Operaciones ISM

- OpenSearch Service es compatible con tres operaciones de ISM únicas, `warm_migration`, `cold_migration` y `cold_delete`:
  - Si su dominio tiene UltraWarm habilitado, la acción `warm_migration` pasa el índice al almacenamiento templado.
  - Si el dominio tiene habilitado el almacenamiento frío, la acción `cold_migration` pasa el índice al almacenamiento frío, y la acción `cold_delete` elimina el índice del almacenamiento frío.

  Aunque una de estas acciones no se complete dentro del periodo de tiempo de espera establecido, la migración o eliminación de los índices prosigue. Configuración de un error_notification de una de las acciones anteriores, le notificará que la acción falla por no haberse completado dentro del periodo de tiempo de espera, pero la notificación es solo para su propia referencia. La propia operación no tiene un tiempo de espera inherente, y continúa ejecutándose hasta que finalmente se completa correctamente o falla.
- Si el dominio ejecuta OpenSearch o Elasticsearch 7.4 o posterior, el servicio OpenSearch admite las operaciones `open` y `close` de ISM.

- Si el dominio ejecuta OpenSearch o Elasticsearch 7.7 o posterior, el servicio OpenSearch admite la operación `snapshot`.

Operaciones de ISM de almacenamiento en frío

Para índices almacenados en frío, debe especificar un parámetro `?type=_cold` cuando utilice las siguientes API de ISM:

- Agregar política
- Quitar política
- Actualizar política

- Reintentar índice fallido
- Explicar índice

Estas API para índices fríos tienen las siguientes diferencias adicionales:

- Los operadores comodín no son compatibles excepto cuando se utilizan al final. Por ejemplo, se admite `_plugins/_ism/<add, remove, change_policy, retry, explain>/logstash-*`, pero no `_plugins/_ism/<add, remove, change_policy, retry, explain>/iad-*-prod`.
- No se admiten varios nombres y patrones de índices. Por ejemplo, se admite `_plugins/_ism/<add, remove, change_policy, retry, explain>/app-logs`, pero no `_plugins/_ism/<add, remove, change_policy, retry, explain>/app-logs,sample-data`.

Configuración de ISM

OpenSearch y Elasticsearch permite cambiar todas las configuraciones de ISM disponibles mediante la API `_cluster/settings`. En Amazon OpenSearch Service, solo puede cambiar la siguiente configuración de ISM:

- Configuración de nivel de clúster:
  - `plugins.index_state_management.enabled`
  - `plugins.index_state_management.history.enabled`
- Configuración de nivel de índice:
  - `plugins.index_state_management.rollover_alias`

Tutorial: Automatización de los procesos de administración del estado del índice

Este tutorial demuestra cómo implementar una política de ISM que automatiza tareas rutinarias de administración de índices y las aplica a índices y patrones de índices.

Administración de estados de índice (ISM) de Amazon OpenSearch Service le permite automatizar las actividades recurrentes de administración de índices para evitar el uso de herramientas adicionales para administrar los ciclos de vida de los índices. Puede crear una política que

automatice estas operaciones en función de la antigüedad del índice, el tamaño y otras condiciones, todo desde su dominio de Amazon OpenSearch Service.

OpenSearch Service admite tres niveles de almacenamiento: el estado “activo” predeterminado para la escritura activa y el análisis de baja latencia, UltraWarm para datos de solo lectura de hasta tres petabytes y el almacenamiento en frío para el archivado ilimitado a largo plazo.

Este tutorial presenta un caso de uso de ejemplo del manejo de datos de serie temporal en índices diarios. En este tutorial, configurará una política que tome una instantánea automática de cada índice adjunto después de 24 horas. A continuación, la política cambia el índice del estado activo predeterminado al almacenamiento UltraWarm después de dos días, al almacenamiento en frío después de 30 días y, por último, elimina el índice después de 60 días.

Requisitos previos

- El dominio de OpenSearch Service debe ejecutar Elasticsearch 6.8 o versiones posteriores.
- Su dominio debe tener almacenamiento UltraWarm y en frío habilitados.
- Debe registrar un repositorio de instantáneas manuales para su dominio.
- Su rol de usuario necesita permisos suficientes para acceder a la consola de OpenSearch Service. Si es necesario, valide y configure el acceso a su dominio.

Paso 1: configurar la política de ISM

Primero, configure una política de ISM en OpenSearch Dashboards.

1. En el panel del dominio en la consola de OpenSearch Service, diríjase a la URL de paneles de OpenSearch Dashboards e inicie sesión con su nombre de usuario maestro y contraseña. La URL tiene este formato: *domain-endpoint*`/_dashboards/`.
2. En OpenSearch Dashboards, seleccione Agregar datos de ejemplo y agregue uno o más de los índices de muestra a su dominio.
3. Abra el panel de navegación izquierdo y elija Administración de índices y, luego, Crear política.
4. Nombre la política `ism-policy-example`.
5. Reemplace la política predeterminada con la siguiente política:

```
{
  "policy": {
    "description": "Move indexes between storage tiers",
    "default_state": "hot",
```

```
    "states": [
      {
        "name": "hot",
        "actions": [],
        "transitions": [
          {
            "state_name": "snapshot",
            "conditions": {
              "min_index_age": "24h"
            }
          }
        ]
      },
      {
        "name": "snapshot",
        "actions": [
          {
            "retry": {
              "count": 5,
              "backoff": "exponential",
              "delay": "30m"
            },
            "snapshot": {
              "repository": "snapshot-repo",
              "snapshot": "ism-snapshot"
            }
          }
        ],
        "transitions": [
          {
            "state_name": "warm",
            "conditions": {
              "min_index_age": "2d"
            }
          }
        ]
      },
      {
        "name": "warm",
        "actions": [
          {
            "retry": {
              "count": 5,
              "backoff": "exponential",
```

```
            "delay": "1h"
          },
          "warm_migration": {}
        }
      ],
      "transitions": [
        {
          "state_name": "cold",
          "conditions": {
            "min_index_age": "30d"
          }
        }
      ]
    },
    {
      "name": "cold",
      "actions": [
        {
          "retry": {
            "count": 5,
            "backoff": "exponential",
            "delay": "1h"
          },
          "cold_migration": {
            "start_time": null,
            "end_time": null,
            "timestamp_field": "@timestamp",
            "ignore": "none"
          }
        }
      ],
      "transitions": [
        {
          "state_name": "delete",
          "conditions": {
            "min_index_age": "60d"
          }
        }
      ]
    },
    {
      "name": "delete",
      "actions": [
        {
```

```
            "cold_delete": {}
          }
        ],
        "transitions": []
      }
    ],
    "ism_template": [
      {
        "index_patterns": [
          "index-*"
        ],
        "priority": 100
      }
    ]
  }
}
```

> **Note**
>
> El campo `ism_template` adjunta automáticamente la política a cualquier índice creado recientemente que coincida con uno de los valores `index_patterns` especificados. En este caso, todos los índices que comiencen por `index-`. Puede modificar este campo para que coincida con un formato de índice de su entorno. Para más información, consulte Plantillas de ISM.

6. En la sección `snapshot` de la política, sustituya *`snapshot-repo`* por el nombre del repositorio de instantáneas que registró para su dominio. También puede optar por sustituir *`ism-snapshot`*, que será el nombre de la instantánea cuando se cree.
7. Seleccione Crear. La política se puede ver ahora en la página Políticas de administración de estados.

Paso 2: adjuntar la política a uno o más índices

Ya que creó la política, puede adjuntarla a uno o más índices de su clúster.

1. Vaya a la pestaña Índices activos y busque `opensearch_dashboards_sample`, donde se enumeran todos los índices de muestra que agregó en el paso 1.

2. Seleccione todos los índices y seleccione Aplicar política y, a continuación, seleccione la política ism-policy-example que acaba de crear.
3. Seleccione Aplicar.

Puede supervisar los índices a medida que pasan por los distintos estados de la página Índices administrados por políticas.

Resumen de índices en Amazon OpenSearch Service con acumulaciones de índices

Las acumulaciones de índices de Amazon OpenSearch Service permiten reducir los costos de almacenamiento mediante la acumulación periódica de datos antiguos en índices resumidos.

Seleccione los campos en los que está interesado y utilice una acumulación de índices para crear un nuevo índice con solo los campos agregados en buckets de tiempo más rigurosos. Puede almacenar meses o años de datos históricos a una fracción del costo con el mismo rendimiento de consulta.

Las acumulaciones de índices requieren OpenSearch o Elasticsearch 7.9 o versiones posteriores.

> **Note**
>
> Esta documentación le ayuda a empezar a crear un trabajo acumulativo de índices en Amazon OpenSearch Service. Para obtener la documentación completa, que incluye una lista con toda la configuración disponible y una referencia completa de la API, consulte Index rollups en la documentación de OpenSearch.

Crear un trabajo acumulativo de índices

Para comenzar, elija Index Management (Gestión de índices) en OpenSearch Dashboards. Seleccione Rollup Jobs (Trabajos de acumulación) y elija Create rollup job (Creación de un trabajo de acumulación).

Paso 1: configuración de índices

Configure los índices de origen y destino. El índice de fuente es el que desea resumir. El índice de destino es donde se guardan los resultados de la acumulación de índices.

Después de crear un trabajo de acumulación de índices, no puede cambiar las selecciones de índices.

Paso 2: definir agregaciones y métricas

Seleccione los atributos con las agregaciones (términos e histogramas) y métricas (promedio, suma, máximo, mínimo y recuento de valores) que desea acumular. Asegúrese de no agregar muchos atributos altamente pormenorizados, ya que no ahorrará mucho espacio.

Paso 3: especificar las programaciones

Especifique una programación para acumular los índices a medida que se ingieren. El trabajo de acumulación de índices está habilitado de forma predeterminada.

Paso 4: revisar y crear

Revise su configuración y seleccione Create (Crear).

Paso 5: buscar en el índice objetivo

Puede utilizar la API `_search` estándar para buscar en el índice de destino. No se puede acceder a la estructura interna de los datos en el índice de destino porque el complemento vuelve a escribir automáticamente la consulta en segundo plano para adaptarse al índice de destino. Esto es para asegurarse de que puede utilizar la misma consulta para el índice de fuente y destino.

Para consultar el índice de destino, establezca `size` en 0:

```
GET target_index/_search
{
  "size": 0,
  "query": {
    "match_all": {}
  },
  "aggs": {
    "avg_cpu": {
      "avg": {
        "field": "cpu_usage"
      }
    }
  }
}
```

> **Note**
>
> Las versiones 2.2 y posteriores de OpenSearch admiten la búsqueda de varios índices acumulativos en una sola solicitud. Las versiones de OpenSearch anteriores a la 2.2 y las versiones de Elasticsearch OSS heredadas solo admiten un índice acumulativo por búsqueda.

Transformación de índices en Amazon OpenSearch Service

Mientras que los trabajos acumulativos de índices le permiten reducir la granularidad de los datos acumulando datos antiguos en índices condensados, los trabajos de transformación le permiten crear una vista resumida diferente de sus datos centrada en ciertos campos, de modo que pueda visualizar o analizar los datos de diferentes maneras.

Las transformaciones de índice tienen una interfaz de usuario de OpenSearch Dashboards y API de REST. La función requiere OpenSearch 1.0 o posterior.

> **Note**
>
> Esta documentación proporciona una breve información general de las transformaciones de índices para ayudarlo a empezar a utilizarlas en un dominio de Amazon OpenSearch Service. Para obtener la documentación completa y una referencia de API de REST, consulte Index transforms en la documentación de código de abierto de OpenSearch.

Creación de un trabajo de transformación de índice

Si no tiene ningún dato en el clúster, use los datos de vuelo de ejemplo de los paneles de OpenSearch para probar trabajos de transformación. Después de agregar los datos, inicie OpenSearch Dashboards. A continuación, elija Administración de índices, Trabajos de transformación, y Crear trabajo de transformación.

Paso 1: elección de índices

En la sección Índices, seleccione el índice de origen y destino. Puede seleccionar un índice de destino existente o crear uno nuevo introduciendo un nombre para él.

Si desea transformar solo un subconjunto de su índice fuente, elija Agregar filtro de datos, y utilice la herramienta de OpenSearch consulta de DSL para especificar un subconjunto de su índice de origen.

Paso 2: Elegir los campos

Después de elegir los índices, elija los campos que desea utilizar en el trabajo de transformación, así como si desea utilizar agrupaciones o agregaciones.

- Puede utilizar agrupaciones para colocar los datos en depósitos separados en el índice transformado. Por ejemplo, si desea agrupar todos los destinos de aeropuerto dentro de los datos de vuelo de muestra, agrupe el `DestAirportID` en un campo de destino de `DestAirportID_terms` y puede encontrar los ID de aeropuerto agrupados en el índice transformado una vez finalizado el trabajo de transformación.
- Por otro lado, las agregaciones le permiten realizar cálculos simples. Por ejemplo, puede incluir una agregación en el trabajo de transformación para definir un nuevo campo de `sum_of_total_ticket_price` que calcula la suma de todos los billetes de avión. A continuación, puede analizar los nuevos datos en su índice transformado.

Paso 3: Especificar un programa

Los trabajos de transformación están habilitados de forma predeterminada y se ejecutan en programaciones. Para transform execution interval (intervalo de ejecución de transformación), especifique un intervalo en minutos, horas o días.

Paso 4: Revisar y supervisar

Revise su configuración y seleccione Crear. A continuación, monitoree la columna Estado del trabajo de transformación.

Paso 5: Buscar en el índice objetivo

Una vez finalizado el trabajo, puede usar la API `_search` para buscar en el índice de destino.

Por ejemplo, después de ejecutar un trabajo de transformación que transforma los datos de vuelo basados en el campo `DestAirportID`, puede ejecutar la siguiente solicitud para devolver todos los campos que tienen un valor de `SFO`:

```
GET target_index/_search
{
  "query": {
```

```
    "match": {
      "DestAirportID_terms" : "SFO"
    }
  }
}
```

Replicación entre clústeres de Amazon OpenSearch Service

Con la replicación entre clústeres de Amazon OpenSearch Service, se pueden replicar índices de usuario, asignaciones y metadatos de un dominio de OpenSearch Service a otro. Utilizar la replicación entre clústeres contribuye a garantizar la posibilidad de recuperación de desastres en caso de interrupción, y permite replicar datos entre centros de datos lejanos geográficamente para reducir la latencia. Usted paga los cargos de estándar de transferencia de datos de AWS para los datos transferidos entre dominios.

La replicación entre clústeres sigue un modelo de replicación activa-pasiva en el que el índice local o seguidor extrae datos del índice remoto o principal. El índice líder hace referencia al origen de los datos o al índice desde el que desea replicar los datos. El índice seguidor hace referencia al destino de los datos o al índice en donde desea replicar los datos.

La replicación entre clústeres está disponible en los dominios que ejecutan Elasticsearch 7.10 o OpenSearch 1.1 o posterior.

> **Note**
>
> En esta documentación se describe cómo configurar la replicación entre clústeres desde la perspectiva de Amazon OpenSearch Service. Esto incluye el uso de la Consola de administración de AWS para configurar conexiones entre clústeres, lo que no es posible en un clúster de OpenSearch autoadministrado. Para obtener la documentación completa, que incluye una referencia de configuración y una referencia completa de la API, consulte Cross-cluster replication en la documentación de OpenSearch.

Temas

- Limitaciones
- Requisitos previos
- Requisitos de los permisos
- Configuración de una conexión entre clústeres

- Inicio de la replicación
- Confirmación de replicación
- Pausa y reanudación de la replicación
- Detención de la replicación
- Seguimiento automático
- Actualización de los dominios conectados

Limitaciones

La replicación entre clústeres tiene las siguientes limitaciones:

- No se pueden replicar datos entre dominios de Amazon OpenSearch Service y clústeres de OpenSearch o Elasticsearch autoadministrados.
- No puede replicar un índice de un dominio seguidor a otro dominio seguidor. Si desea replicar un índice en varios dominios seguidores, solo puede replicarlo desde el dominio líder único.
- Se puede conectar un dominio, mediante una combinación de conexiones de entrada y salida, a un máximo de 20 dominios más.
- Al configurar inicialmente una conexión entre clústeres, el dominio principal debe tener la misma versión o una versión superior a la del dominio seguidor.
- No puede usar CloudFormation para conectar dominios.
- No se puede utilizar la replicación entre clústeres en instancias M3 o bursátiles (T2 y T3).
- No puede replicar datos entre índices UltraWarm o frío. Ambos índices deben estar almacenados en caliente.
- Al eliminar un índice del dominio principal, no se elimina automáticamente el índice correspondiente en el dominio seguidor.

Requisitos previos

Antes de configurar la replicación entre clústeres, asegúrese de que los dominios cumplan los siguientes requisitos:

- Elasticsearch 7.10 u OpenSearch 1.1 o posterior
- Control de acceso detallado habilitado
- Cifrado de nodo a nodo habilitado

Requisitos de los permisos

Para iniciar la replicación, debe incluir el permiso `es:ESCrossClusterGet` en el dominio remoto (líder). Recomendamos la siguiente política de IAM en el dominio remoto. Esta política también permite realizar otras operaciones, como indexar documentos y efectuar búsquedas estándar:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "*"
        ]
      },
      "Action": [
        "es:ESHttp*"
      ],
      "Resource": "arn:aws:es:us-east-1:111122223333:domain/leader-domain/*"
    },
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
      },
      "Action": "es:ESCrossClusterGet",
      "Resource": "arn:aws:es:us-east-1:111122223333:domain/leader-domain"
    }
  ]
}
```

Asegúrese de que el permiso `es:ESCrossClusterGet` se aplica para `/leader-domain` y no `/leader-domain/*`.

Para que los usuarios que no son administradores puedan realizar actividades de replicación, también se les deben asignar los permisos adecuados. La mayoría de los permisos corresponden a operaciones de API REST específicas. Por ejemplo, el permiso `indices:admin/plugins/`

`replication/index/_resume` permite reanudar la replicación de un índice. Para obtener una lista completa de permisos, consulte Permisos de replicación en la documentación de OpenSearch.

> **Note**
>
> Los comandos para iniciar la replicación y crear una regla de replicación son casos especiales. Dado que invocan procesos en segundo plano en los dominios líder y seguidor, se debe pasar un elemento `leader_cluster_role` y `follower_cluster_role` en la solicitud. OpenSearch Service utiliza estos roles en todas las tareas de replicación de backend. Para obtener más información sobre la asignación y el uso de estos roles, consulte Asignación de los roles de clúster líder y seguidor en la documentación de OpenSearch.

Configuración de una conexión entre clústeres

Para replicar índices de un dominio a otro, se debe configurar una conexión entre clústeres entre los dominios. La forma más sencilla de conectar dominios es a través de la pestaña Connections (Conexiones) del panel del dominio. También puede usar la API de configuración o la AWS CLI. Dado que la replicación entre clústeres sigue un modelo de "extracción", las conexiones se inician desde el dominio seguidor.

> **Note**
>
> Si se han conectado anteriormente dos dominios para realizar búsquedas entre clústeres, no se puede usar la misma conexión para la replicación. La conexión está marcada como `SEARCH_ONLY` en la consola. Para realizar la replicación entre dos dominios conectados previamente, se debe eliminar la conexión y volver a crearla. Una vez que se haya hecho esto, la conexión queda disponible tanto para la búsqueda entre clústeres como para la replicación entre clústeres.

Para configurar una conexión

1. En la consola de Amazon OpenSearch Service, seleccione el dominio seguidor, vaya a la pestaña Connections (Conexiones) y elija Request (Solicitar).
2. En Alias de conexión, ingrese un nombre para la conexión.
3. Seleccione entre conectarse a un dominio de su Cuenta de AWS y región, o hacerlo a otra cuenta o región.

- Para conectarse a un dominio de su Cuenta de AWS y región, seleccione el dominio y elija Solicitar.
- Para conectarse a un dominio de otra Cuenta de AWS o región, especifique el ARN del dominio remoto y elija Solicitar.

OpenSearch Service valida la solicitud de conexión. Si los dominios son incompatibles, se produce un error en la conexión. Si la validación se realiza correctamente, se envía al dominio de destino para su aprobación. Cuando el dominio de destino aprueba la solicitud, puede iniciar la replicación.

La replicación entre clústeres admite replicación bidireccional. Esto significa que puede crear una conexión saliente del dominio A al dominio B y otra conexión saliente del dominio B al dominio A. A continuación, puede configurar la replicación para que el dominio A siga un índice en el dominio B y el dominio B siga un índice en el dominio A.

Inicio de la replicación

Después de establecer una conexión entre clústeres, puede empezar a replicar datos. En primer lugar, cree un índice en el dominio líder que se va a replicar:

```
PUT leader-01
```

Para replicar ese índice, envíe este comando al dominio seguidor:

```
PUT _plugins/_replication/follower-01/_start
{
   "leader_alias": "connection-alias",
   "leader_index": "leader-01",
   "use_roles":{
      "leader_cluster_role": "all_access",
      "follower_cluster_role": "all_access"
   }
}
```

Puede encontrar el alias de conexión en la pestaña Conexiones del panel de control del dominio.

Para simplificar, en este ejemplo se supone que un administrador emite la solicitud y utiliza `all_access` para los elementos `leader_cluster_role` y `follower_cluster_role`. No obstante, en entornos de producción, se recomienda crear usuarios de replicación en los índices

líder y seguidor, y asignarlos como corresponda. Los nombres de usuario deben ser idénticos. Para obtener información sobre estos roles y cómo asignarlos, consulte Asignación de los roles de clúster líder y seguidor en la documentación de OpenSearch.

Confirmación de replicación

Para confirmar que se está produciendo la replicación, obtenga su estado:

```
GET _plugins/_replication/follower-01/_status

{
  "status" : "SYNCING",
  "reason" : "User initiated",
  "leader_alias" : "connection-alias",
  "leader_index" : "leader-01",
  "follower_index" : "follower-01",
  "syncing_details" : {
    "leader_checkpoint" : -5,
    "follower_checkpoint" : -5,
    "seq_no" : 0
  }
}
```

Los valores de los puntos de control líder y seguidor comienzan como enteros negativos y reflejan el número de particiones que se tienen (-1 para una partición, -5 para cinco particiones, etc.). Los valores se incrementan en enteros positivos con cada cambio que se realiza. Si los valores son los mismos, eso significa que los índices están totalmente sincronizados. Puede utilizar estos valores de punto de control para medir la latencia de replicación en todos los dominios.

Para validar aún más la replicación, agregue un documento al índice líder:

```
PUT leader-01/_doc/1
{
   "Doctor Sleep":"Stephen King"
}
```

Y confirme que aparece en el índice seguidor:

```
GET follower-01/_search

{
```

```
    ...
    "max_score" : 1.0,
    "hits" : [
      {
        "_index" : "follower-01",
        "_type" : "_doc",
        "_id" : "1",
        "_score" : 1.0,
        "_source" : {
          "Doctor Sleep" : "Stephen King"
        }
      }
    ]
  }
}
```

Pausa y reanudación de la replicación

Puede pausar temporalmente la replicación si necesita corregir problemas o reducir la carga en el dominio líder. Envíe esta solicitud al dominio seguidor. Asegúrese de incluir un cuerpo de la solicitud vacío:

```
POST _plugins/_replication/follower-01/_pause
{}
```

Después, obtenga el estado para asegurarse de que la replicación esté en pausa:

```
GET _plugins/_replication/follower-01/_status

{
  "status" : "PAUSED",
  "reason" : "User initiated",
  "leader_alias" : "connection-alias",
  "leader_index" : "leader-01",
  "follower_index" : "follower-01"
}
```

Cuando termine de realizar los cambios, reanude la replicación. Envíe esta solicitud al dominio seguidor. Asegúrese de incluir un cuerpo de la solicitud vacío:

```
POST _plugins/_replication/follower-01/_resume
```

```
{}
```

No puede reanudar la replicación después de haber estado en pausa durante más de 12 horas. Debe detener la replicación, eliminar el índice de seguidores y reiniciar la replicación del líder.

Detención de la replicación

Cuando se detiene la replicación por completo, el índice seguidor deja de seguir al líder y se convierte en un índice estándar. No se puede reiniciar la replicación después de detenerla.

Detenga la replicación desde el dominio seguidor. Asegúrese de incluir un cuerpo de la solicitud vacío:

```
POST _plugins/_replication/follower-01/_stop
{}
```

Seguimiento automático

Puede definir un conjunto de reglas de replicación en un único dominio líder que replique automáticamente los índices que coincidan con un patrón especificado. Cuando un índice del dominio líder coincide con uno de los patrones (por ejemplo, `books*`), se crea un índice seguidor coincidente en el dominio seguidor. OpenSearch Service replica todos los índices existentes que coincidan con el patrón, así como los nuevos índices que se creen. No replica los índices que ya existan en el dominio seguidor.

Para replicar todos los índices (a excepción de los índices creados por el sistema y los que ya existan en el dominio seguidor), utilice un patrón comodín (*).

Creación de una regla de replicación

Cree una regla de replicación en el dominio seguidor y especifique el nombre de la conexión entre clústeres:

```
POST _plugins/_replication/_autofollow
{
   "leader_alias" : "connection-alias",
   "name": "rule-name",
   "pattern": "books*",
   "use_roles":{
      "leader_cluster_role": "all_access",
```

```
      "follower_cluster_role": "all_access"
   }
}
```

Puede encontrar el alias de conexión en la pestaña Conexiones del panel de control del dominio.

Para simplificar, en este ejemplo se supone que un administrador emite la solicitud, que utiliza `all_access` como reglas de los dominios líder y seguidor. No obstante, en entornos de producción se recomienda crear usuarios de replicación en los índices líder y seguidor, y asignarlos como corresponda. Los nombres de usuario deben ser idénticos. Para obtener información sobre estos roles y cómo asignarlos, consulte Asignación de los roles de clúster líder y seguidor en la documentación de OpenSearch.

Para recuperar una lista de reglas de replicación existentes en un dominio, utilice la operación de la API de estadísticas de seguimiento automático.

Para probar la regla, cree un índice que coincida con el patrón del dominio líder:

```
PUT books-are-fun
```

Y verifique que su réplica aparece en el dominio seguidor:

```
GET _cat/indices

health status index          uuid                   pri rep docs.count docs.deleted
 store.size pri.store.size
green  open   books-are-fun  ldfHO78xYYdxRMULuiTvSQ     1   1          0            0
     208b           208b
```

Eliminación de una regla de replicación

Cuando se elimina una regla de replicación, OpenSearch Service deja de replicar nuevos índices que coincidan con el patrón, pero prosigue con la actividad de replicación existente hasta que se detenga la replicación de esos índices.

Detenga las reglas de replicación desde el dominio seguidor.

```
DELETE _plugins/_replication/_autofollow
{
   "leader_alias" : "connection-alias",
   "name": "rule-name"
```

```
}
```

Actualización de los dominios conectados

Para actualizar la versión del motor de dos dominios que tienen una conexión entre clústeres, actualice primero el dominio seguidor y, después, el dominio principal. No elimine la conexión entre ellos; de lo contrario, la replicación se detendrá y no podrá reanudarla.

Migración de índices de Amazon OpenSearch Service mediante la reindexación remota

La reindexación remota le permite copiar índices de un dominio de Amazon OpenSearch Service a otro. Puede migrar índices desde cualquier dominio de OpenSearch Service o clúster autoadministrado de OpenSearch y Elasticsearch.

Un dominio y un índice remotos hacen referencia al origen de los datos o al dominio y el índice desde los que desea copiar los datos. Un dominio y un índice locales hacen referencia al destino de los datos o al dominio y el índice en donde desea copiar los datos.

La reindexación remota requiere OpenSearch 1.0 o posterior, o Elasticsearch 6.7 o posterior, en el dominio local. El dominio remoto debe ser inferior o la misma versión principal que el dominio local. Las versiones de Elasticsearch se consideran inferiores que las versiones de OpenSearch; esto significa que puede reindexar los datos de los dominios de Elasticsearch a los dominios de OpenSearch. Dentro de la misma versión principal, el dominio remoto puede ser cualquier versión secundaria. Por ejemplo, se admite la reindexación remota de Elasticsearch 7.10.x a 7.9, pero OpenSearch 1.0 a Elasticsearch 7.10.x no es compatible.

> **Note**
>
> En esta documentación se describe cómo reindexar datos entre dominios de Amazon OpenSearch Service. Para obtener la documentación completa sobre la operación `reindex`, lo que incluye pasos detallados y opciones admitidas, consulte Reindex document en la documentación de OpenSearch.

Temas

- Requisitos previos

- Reindexación de datos entre dominios de internet de OpenSearch Service
- Reindexación de datos entre dominios de OpenSearch Service cuando el remoto está en una VPC
- Reindexar datos entre dominios que no son de OpenSearch Service
- Reindexar conjuntos de datos grandes
- Configuración remota de reindexación

Requisitos previos

La reindexación remota requiere lo siguiente:

- El dominio remoto debe ser accesible desde el dominio local. Para un dominio remoto que reside en una VPC, el dominio local debe tener acceso a la VPC. Este proceso depende de la configuración de la red, pero probablemente implica conectarse a una VPN o a una red administrada, o utilizar la conexión de punto de conexión de VPC nativa. Para obtener más información, consulte the section called “Compatibilidad con VPC”.
- La solicitud debe autorizarse por el dominio remoto como cualquier otra solicitud REST. Si el dominio remoto tiene habilitado el control de acceso detallado, debe tener permiso para realizar la reindexación en el dominio remoto y leer el índice en el dominio local. Para obtener más consideraciones de seguridad, consulte the section called “Control de acceso detallado”.
- Recomendamos que cree un índice con la configuración deseada en el dominio local antes de comenzar con el proceso de reindexación.
- Si un dominio utiliza un tipo de instancia T2 o T3 para los nodos de datos, no se puede utilizar la reindexación en remoto.

Reindexación de datos entre dominios de internet de OpenSearch Service

El escenario más básico es que el índice remoto se encuentra en el mismo Región de AWS que el dominio local con un punto de conexión de acceso público y que firmó credenciales de IAM.

Desde el dominio remoto, especifique el índice remoto desde el que se va a reindexar y el índice local para reindexar:

```
POST _reindex
{
  "source": {
    "remote": {
```

```
      "host": "https://remote-domain-endpoint:443"
    },
    "index": "remote_index"
  },
  "dest": {
    "index": "local_index"
  }
}
```

Debe agregar 443 al final del punto de conexión del dominio remoto para una comprobación de validación.

Para comprobar que el índice se ha copiado en el dominio local, envíe esta solicitud al dominio local:

```
GET local_index/_search
```

Si el índice remoto se encuentra en una región diferente del dominio local, pase su nombre de región, como en esta solicitud de ejemplo:

```
POST _reindex
{
  "source": {
    "remote": {
      "host": "https://remote-domain-endpoint:443",
      "region": "eu-west-1"
    },
    "index": "remote_index"
  },
  "dest": {
    "index": "local_index"
  }
}
```

En el caso de regiones aisladas como AWS GovCloud (US) o China, es posible que no se pueda acceder al punto de conexión porque el usuario de IAM no se reconoce en esas regiones.

Si el dominio remoto está protegido con autenticación básica, especifique el nombre de usuario y la contraseña:

```
POST _reindex
{
  "source": {
```

```
    "remote": {
      "host": "https://remote-domain-endpoint:443",
      "username": "username",
      "password": "password"
    },
    "index": "remote_index"
  },
  "dest": {
    "index": "local_index"
  }
}
```

Reindexación de datos entre dominios de OpenSearch Service cuando el remoto está en una VPC

Cada dominio de OpenSearch Service se compone de su propia infraestructura de nube privada virtual (VPC) interna. Cuando se crea un nuevo dominio en una VPC de OpenSearch Service existente, se crea una interfaz de red elástica para cada nodo de datos de la VPC.

Dado que la operación de reindexación remota se realiza desde el dominio remoto de OpenSearch Service y, por tanto, dentro de su propia VPC privada, se necesita una forma de acceder a la VPC del dominio local. Puede hacerlo mediante la característica de conexión de punto de conexión de VPC integrada para establecer una conexión a través de AWS PrivateLink o configurando un proxy.

Si el dominio local utiliza la versión 1.0 o posterior de OpenSearch, se puede usar la consola o la AWS CLI para crear una conexión de AWS PrivateLink. Una conexión de AWS PrivateLink permite que los recursos de la VPC local se conecten de manera privada a los recursos de la VPC remota dentro del mismo Región de AWS.

Para crear una conexión de punto de conexión de VPC, el dominio de origen que se va a reindexar debe estar en una VPC local y tanto el dominio de origen como el de destino deben estar en la misma Región de AWS.

Reindexación de los datos con la Consola de administración de AWS

Puede usar la reindexación remota con la consola para copiar índices entre dos dominios que comparten una conexión de punto de conexión de VPC.

1. Vaya a la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/.
2. En el panel de navegación izquierdo, seleccione Dominios.

3. Seleccione el dominio local, que es el dominio en el que desea copiar los datos. Se abrirá la página de detalles del dominio. Seleccione la pestaña Conexiones situada debajo de la información general, y luego Solicitar.
4. En la página Solicitar conexión, seleccione Conexión de punto de conexión de VPC como modo de conexión e ingrese otros detalles relevantes. Estos detalles incluyen el dominio remoto, que es el dominio desde el que desea copiar los datos. A continuación, seleccione Solicitar.
5. Vaya a la página de detalles del dominio remoto, seleccione la pestaña Conexiones y busque la tabla Conexiones entrantes. Seleccione la casilla de verificación situada junto al nombre del dominio desde el que acaba de crear la conexión (el dominio local). Seleccione Aprobar.
6. Vuelva al dominio local, seleccione la pestaña Conexiones y busque la tabla Conexiones salientes. Una vez que la conexión entre los dos dominios esté activa, aparecerá un punto de conexión disponible en la columna Punto de conexión de la tabla. Copie el punto de conexión.
7. Abra el panel de control del dominio local y seleccione Herramientas de desarrollo en el panel de navegación de la izquierda. Para confirmar que el índice del dominio remoto aún no existe en el dominio local, ejecute la siguiente solicitud GET. Sustituya *remote-domain-index-name* por su propio nombre de índice.

   ```
   GET remote-domain-index-name/_search
   {
      "query":{
         "match_all":{}
      }
   }
   ```

   En el resultado, debería aparecer un error que indica que no se ha encontrado el índice.
8. Debajo de la solicitud GET, cree una solicitud POST y utilice el punto de conexión como host remoto, tal y como sigue.

   ```
   POST _reindex
   {
      "source":{
         "remote":{
            "host":"connection-endpoint",
            "username":"username",
            "password":"password"
         },
         "index":"remote-domain-index-name"
      },
   ```

```
   "dest":{
      "index":"local-domain-index-name"
   }
}
```

Ejecute esta solicitud.

9. Vuelva a ejecutar la solicitud GET. Ahora el resultado debería indicar que el índice local existe. Puede consultar este índice para comprobar que OpenSearch haya copiado todos los datos del índice remoto.

Reindexación de los datos con las operaciones de OpenSearch Service de OpenSearch API de OpenSearch

Puede usar la reindexación remota con la API para copiar índices entre dos dominios que comparten una conexión de punto de conexión de VPC.

1. Use la operación de API CreateOutboundConnection para solicitar una nueva conexión desde el dominio local al dominio remoto.

```
POST https://es.region.amazonaws.com/2021-01-01/opensearch/cc/outboundConnection

{
   "ConnectionAlias": "remote-reindex-example",
   "ConnectionMode": "VPC_ENDPOINT",
   "LocalDomainInfo": {
      "AWSDomainInformation": {
         "DomainName": "local-domain-name",
         "OwnerId": "aws-account-id",
         "Region": "region"
      }
   },
   "RemoteDomainInfo": {
      "AWSDomainInformation": {
         "DomainName": "remote-domain-name",
         "OwnerId": "aws-account-id",
         "Region": "region"
      }
   }
}
```

Recibirá un `ConnectionId` en la respuesta. Guarde este ID para utilizarlo en el siguiente paso.

2. Use la operación de la API AcceptInboundConnection con su ID de conexión para aprobar la solicitud del dominio local.

```
PUT https://es.region.amazonaws.com/2021-01-01/opensearch/cc/
inboundConnection/ConnectionId/accept
```

3. Use la operación de API DescribeOutboundConnections para recuperar el punto de conexión de su dominio remoto.

```
{
    "Connections": [
        {
            "ConnectionAlias": "remote-reindex-example",
            "ConnectionId": "connection-id",
            "ConnectionMode": "VPC_ENDPOINT",
            "ConnectionProperties": {
                "Endpoint": "connection-endpoint"
            },
            ...
        }
    ]
}
```

Guarde el *connection-endpoint* para usarlo en el paso 5.

4. Para confirmar que el índice del dominio remoto aún no existe en el dominio local, ejecute la siguiente solicitud GET. Sustituya *remote-domain-index-name* por su propio nombre de índice.

```
GET local-domain-endpoint/remote-domain-index-name/_search
{
   "query":{
      "match_all":{}
   }
}
```

En el resultado, debería aparecer un error que indica que no se ha encontrado el índice.

5. Cree una solicitud POST y utilice el punto de conexión como host remoto, tal y como se muestra a continuación.

```
POST local-domain-endpoint/_reindex
{
   "source":{
      "remote":{
         "host":"connection-endpoint",
         "username":"username",
         "password":"password"
      },
      "index":"remote-domain-index-name"
   },
   "dest":{
      "index":"local-domain-index-name"
   }
}
```

Ejecute esta solicitud.

6. Vuelva a ejecutar la solicitud GET. Ahora el resultado debería indicar que el índice local existe. Puede consultar este índice para comprobar que OpenSearch haya copiado todos los datos del índice remoto.

Si el dominio remoto está alojado dentro de una VPC y no desea utilizar la característica de conexión de punto de conexión de VPC, debe configurar un proxy con un punto de conexión accesible públicamente. En este caso, OpenSearch Service requiere un punto de conexión público porque no tiene la capacidad de enviar tráfico a su VPC.

Cuando ejecuta un dominio en modo VPC, se colocan uno o más puntos de conexión en su VPC. Sin embargo, estos puntos de conexión son únicamente para el tráfico que entra en el dominio de la VPC y no permiten que el tráfico entre en la VPC misma.

El comando reindexación remota se ejecuta desde el dominio local, por lo que el tráfico de origen no puede usar esos puntos de conexión para acceder al dominio remoto. Por eso se requiere un proxy en este caso de uso. El dominio proxy debe tener un certificado firmado por una autoridad de certificación (CA) pública. No se admiten los certificados autofirmados o firmados por una entidad de certificación privada.

Reindexar datos entre dominios que no son de OpenSearch Service

Si el índice remoto está alojado fuera de OpenSearch Service, como en una instancia de EC2 autoadministrada, establezca el parámetro `external` en `true`:

```
POST _reindex
{
  "source": {
    "remote": {
      "host": "https://remote-domain-endpoint:443",
      "username": "username",
      "password": "password",
      "external": true
    },
    "index": "remote_index"
  },
  "dest": {
    "index": "local_index"
  }
}
```

En este caso, solo se admite la autenticación básica con un nombre de usuario y contraseña. El dominio remoto debe tener un punto de conexión de acceso público (incluso si está en la misma VPC que el dominio de OpenSearch Service local) y un certificado firmado por una CA pública. No se admiten los certificados autofirmados o firmados por una CA privada.

Reindexar conjuntos de datos grandes

La reindexación remota envía una solicitud de desplazamiento al dominio remoto con los siguientes valores predeterminados:

- Contexto de búsqueda de 5 minutos
- Tiempo de espera del socket de 30 segundos
- Tamaño del lote de 1000

Recomendamos ajustar estos parámetros para adaptarlos a sus datos. Para documentos grandes, considere un tamaño de lote más pequeño o un tiempo de espera más largo. Para obtener más información, consulte Paginación de resultados.

```
POST _reindex?pretty=true&scroll=10h&wait_for_completion=false
{
  "source": {
    "remote": {
      "host": "https://remote-domain-endpoint:443",
```

```
      "socket_timeout": "60m"
    },
    "size": 100,
    "index": "remote_index"
  },
  "dest": {
    "index": "local_index"
  }
}
```

También recomendamos agregar la siguiente configuración al índice local para un mejor rendimiento:

```
PUT local_index
{
  "settings": {
    "refresh_interval": -1,
    "number_of_replicas": 0
  }
}
```

Una vez finalizado el proceso de reindexación, puede establecer el recuento de réplicas deseado y eliminar la configuración del intervalo de actualización.

Para reindexar únicamente un subconjunto de documentos que seleccione a través de una consulta, envíe esta solicitud al dominio local:

```
POST _reindex
{
  "source": {
    "remote": {
      "host": "https://remote-domain-endpoint:443"
    },
    "index": "remote_index",
    "query": {
      "match": {
        "field_name": "text"
      }
    }
  },
  "dest": {
    "index": "local_index"
  }
```

```
}
```

La reindexación remota no admite la división, por lo que no puede realizar varias operaciones de desplazamiento para la misma solicitud en paralelo.

Configuración remota de reindexación

Además de las opciones estándar de reindexación, OpenSearch Service admite las siguientes opciones:

Opciones	Valores válidos	Descripción	Obligatorio
externo	Booleano	Si el dominio remoto no es un dominio de OpenSearch Service, o si está reindexando entre dos dominios VPC, especifique como `true`.	No
region	Cadena	Si el dominio remoto se encuentra en otra región, especifique el nombre de la región.	No

Administración de datos de series temporales en Amazon OpenSearch Service con Data Streams

Un flujo de trabajo típico para administrar datos de serie temporal consta de varios pasos, como la creación de un alias de índice de reversión, la definición de un índice de escritura y la definición de mapeos y configuraciones comunes para los índices de respaldo.

Data Streams de Amazon OpenSearch Service ayuda a simplificar este proceso de configuración inicial. Data Streams funciona de inmediato para los datos basados en el tiempo, como los registros de aplicaciones, que normalmente son anexos.

Data Streams requiere la versión 1.0 o posterior de OpenSearch.

> **Note**
>
> Esta documentación proporciona los pasos básicos que le ayudarán a empezar a utilizar los flujos de datos en un dominio de Amazon OpenSearch Service. Para obtener documentación completa, consulte Data streams en la documentación de OpenSearch.

Introducción a Data Streams

Un flujo de datos se compone internamente de múltiples índices de respaldo. Las peticiones de búsqueda se dirigen a todos los índices de respaldo, mientras que las solicitudes de indexación se dirigen al índice de escritura más reciente.

Paso 1: crear una plantilla de índices

Para crear un flujo de datos, primero debe crear una plantilla de índice que configure un conjunto de índices como un flujo de datos. El objeto `data_stream` indica que se trata de un flujo de datos y no de una plantilla de índice normal. El patrón de índice coincide con el nombre del flujo de datos:

```
PUT _index_template/logs-template
{
  "index_patterns": [
    "my-data-stream",
    "logs-*"
  ],
  "data_stream": {},
  "priority": 100
}
```

En este caso, cada documento capturado debe tener un campo `@timestamp`. También puede definir su propio campo de marca temporal personalizado como una propiedad en el objeto `data_stream`.

```
PUT _index_template/logs-template
{
  "index_patterns": "my-data-stream",
  "data_stream": {
    "timestamp_field": {
      "name": "request_time"
    }
  }
```

```
}
```

Paso 2: crear un flujo de datos

Después de crear una plantilla de índice, puede comenzar directamente a incorporar datos sin crear un flujo de datos.

Debido a que tenemos una plantilla de índice coincidente con un objeto `data_stream`, OpenSearch crea automáticamente el flujo de datos:

```
POST logs-staging/_doc
{
  "message": "login attempt failed",
  "@timestamp": "2013-03-01T00:00:00"
}
```

Paso 3: incorporar datos en el flujo de datos

Para incorporar datos en un flujo de datos, puede utilizar las API de indexación regulares. Asegúrese de que cada documento que indexe tenga un campo temporal. Si intenta incorporar un documento que no tiene un campo temporal, aparecerá un error.

```
POST logs-redis/_doc
{
  "message": "login attempt",
  "@timestamp": "2013-03-01T00:00:00"
}
```

Paso 4: búsqueda de un flujo de datos

Puede buscar en un flujo de datos igual que en un índice normal o en un alias de índice. La operación de búsqueda se aplica a todos los índices de respaldo (todos los datos presentes en el flujo).

```
GET logs-redis/_search
{
  "query": {
    "match": {
      "message": "login"
    }
```

```
  }
}
```

Paso 5: revertir un flujo de datos

Puede configurar una política de Administración de estados de índice (ISM) para automatizar el proceso de reversión del flujo de datos. La política de ISM se aplica a los índices de respaldo en el momento de su creación. Cuando se asocia una política a un flujo de datos, solo afecta a los índices de respaldo futuros de ese flujo de datos. Tampoco es necesario proporcionar la configuración `rollover_alias`, ya que la política de ISM deduce esta información del índice de respaldo.

> **Note**
>
> Si migra un índice de respaldo al almacenamiento en frío, OpenSearch elimina este índice del flujo de datos. Incluso si vuelve a mover el índice a UltraWarm, el índice sigue siendo independiente y no parte del flujo de datos. Una vez que se haya eliminado un índice del flujo de datos, al buscar en el flujo no se devolverá ningún dato del índice.

> **Warning**
>
> El índice de escritura de un flujo de datos no se puede migrar a un almacenamiento en frío. Si desea migrar los datos de su flujo de datos a un almacenamiento en frío, debe transferir el flujo de datos antes de la migración.

Paso 6: administrar Data Streams en OpenSearch Dashboards

Para administrar Data Streams desde OpenSearch Dashboards, abra OpenSearch Dashboards, seleccione Gestión de índices y, a continuación, seleccione Índices o Índices administrados por políticas.

Paso 7: eliminar un flujo de datos

La operación de eliminación elimina primero los índices de respaldo de un flujo de datos y, a continuación, elimina el propio flujo de datos.

Para eliminar un flujo de datos y todos sus índices de respaldo ocultos:

```
DELETE _data_stream/name_of_data_stream
```

Monitoreo de datos en Amazon OpenSearch Service

Monitoree de forma proactiva sus datos en Amazon OpenSearch Service con alertas y detección de anomalías. Configure alertas para recibir notificaciones cuando los datos superan umbrales determinados. La detección de anomalías utiliza el machine learning para detectar automáticamente cualquier valor atípico en los datos de streaming. Puede emparejar la detección de anomalías con las alertas para asegurarse de que se le notifique tan pronto como se detecte una anomalía.

Temas

- Configuración de alertas en Amazon OpenSearch Service
- Detección de anomalías en Amazon OpenSearch Service

Configuración de alertas en Amazon OpenSearch Service

Configure alertas en Amazon OpenSearch Service para recibir notificaciones cuando los datos de uno o varios índices cumplen determinadas condiciones. Por ejemplo, es posible que desee recibir un email si la aplicación registra más de cinco errores HTTP 503 en una hora o tal vez desee avisar a un desarrollador si no se indexaron nuevos documentos en los últimos 20 minutos.

Las alertas requieren OpenSearch o Elasticsearch 6.2 o posterior.

> **Note**
>
> Esta documentación proporciona una breve información general de las alertas y destaca en qué se diferencian las alertas en un dominio de Amazon OpenSearch Service de las alertas en un clúster de OpenSearch de código abierto. Para obtener la documentación completa sobre alertas, lo que incluye una referencia de la API exhaustiva, una lista de los campos de solicitud disponibles para monitores compuestos y descripciones de variables de acciones y desencadenadores válidos, consulte Alerting en la documentación de OpenSearch.

Temas

- Permisos de alertas
- Introducción a las alertas
- Notificaciones
- Diferencias

Permisos de alertas

Las alertas admiten el control de acceso detallado. Para obtener más información sobre la combinación y la coincidencia de permisos para adaptarlos al caso de uso, consulte Alertas de seguridad en la documentación de OpenSearch.

Para acceder a la página alertas de OpenSearch Dashboards, debe estar asignado al menos al rol `alerting_read_access` predefinido o disponer de permisos equivalentes. Este rol concede permisos para ver alertas, destinos y monitores, pero no para reconocer alertas ni modificar destinos o monitores.

Introducción a las alertas

Para crear una alerta, debe configurar un monitor, que es una tarea que se ejecuta según un cronograma definido y consulta los índices de OpenSearch. También puede configurar disparadores, que definen las condiciones que generan los eventos. Por último, se configuran las acciones, lo que ocurre después de que se activa una alerta.

Para comenzar a trabajar con alertas

1. Seleccione Alerting (Alertas) en el menú principal de OpenSearch Dashboards y, a continuación, Seleccione Create monitor (Crear monitor).
2. Cree un monitor por consulta, por bucket, por métricas de clúster o por documento. Para obtener instrucciones, consulte Create a monitor (Crear un monitor).
3. En Triggers (Disparadores), cree uno o más disparadores. Para obtener instrucciones, consulte Create triggers (Creación de disparadores).
4. En Actions (Acciones), configure un notification channel (canal de notificación) para la alerta. Seleccione entre Slack, Amazon Chime, un webhook personalizado o Amazon SNS. Como puede imaginar, las notificaciones requieren conectividad con el canal. Por ejemplo, para notificar a un canal de Slack o enviar un webhook personalizado a un servidor de terceros, su dominio de OpenSearch Service necesita la conectividad de red apropiada. El webhook personalizado debe tener una dirección IP pública para que un dominio de OpenSearch Service pueda enviarle alertas.

> **Tip**
>
> Después de que una acción envíe correctamente un mensaje, tiene la responsabilidad de asegurar el acceso a ese mensaje (por ejemplo, el acceso a un canal de Slack). Si

el dominio contiene información confidencial, considere utilizar desencadenadores sin acciones y verificar periódicamente si hay alertas en Dashboards.

Notificaciones

Las alertas se integran con Notifications (Notificaciones), que es un sistema unificado para las notificaciones de OpenSearch. Las notificaciones le permiten configurar el servicio de comunicación que desea utilizar y ver las estadísticas pertinentes y la información para la solución de problemas. Para obtener documentación completa, consulte Notifications (Notificaciones) en la documentación de OpenSearch.

Su dominio debe ejecutarse en una versión de OpenSearch 2.3 o posterior para utilizar notificaciones.

> **Note**
>
> Las notificaciones de OpenSearch son independientes de las notificaciones de OpenSearch Service, que proporcionan detalles sobre las actualizaciones del software del servicio, las mejoras de ajuste automático y otra información importante a nivel de dominio. Las notificaciones de OpenSearch son específicas del complemento.

Los canales de notificación sustituyeron a los destinos de alerta a partir de la versión 2.0 de OpenSearch. Los destinos quedaron obsoletos de forma definitiva y, a partir de ahora, todas las notificaciones de alertas se gestionarán a través de los canales.

Al actualizar sus dominios a la versión 2.3 o posterior (dado que la compatibilidad de OpenSearch Service para la versión 2.x comienza con la versión 2.3), sus destinos existentes se migran de forma automática a los canales de notificación. Si un destino no se migra, el monitor seguirá usándolo hasta que el monitor se migre a un canal de notificación. Para obtener más información, consulte Questions about destinations (Preguntas sobre destinos) en la documentación de OpenSearch.

Para empezar con las notificaciones, inicie sesión en OpenSearch Dashboards y seleccione Notifications (Notificaciones), Channels (Canales) y Create channel (Crear canal).

Amazon Simple Notification Service (Amazon SNS) es un tipo de canal compatible para las notificaciones. Para autenticar a los usuarios, debe proporcionar al usuario acceso completo a

Amazon SNS o dejar que asuma un rol de IAM con permisos para acceder a Amazon SNS. Para obtener instrucciones, consulte Amazon SNS as a channel type (Amazon SNS como un tipo de canal).

Diferencias

En comparación con la versión de código abierto de OpenSearch, las alertas en Amazon OpenSearch Service presentan algunas diferencias notables.

Configuración de alertas

OpenSearch Service permite modificar las siguientes Configuraciones de alertas:

- `plugins.scheduled_jobs.enabled`
- `plugins.alerting.alert_history_enabled`
- `plugins.alerting.alert_history_max_age`
- `plugins.alerting.alert_history_max_docs`
- `plugins.alerting.alert_history_retention_period`
- `plugins.alerting.alert_history_rollover_period`
- `plugins.alerting.filter_by_backend_roles`

Todas las demás configuraciones utilizan los valores predeterminados que no se pueden cambiar.

Para deshabilitar las alertas, envíe la siguiente solicitud:

```
PUT _cluster/settings
{
  "persistent" : {
    "plugins.scheduled_jobs.enabled" : false
  }
}
```

La siguiente solicitud configura las alertas para eliminar automáticamente los índices del historial después de siete días, en lugar de los 30 días predeterminados:

```
PUT _cluster/settings
{
  "persistent": {
```

```
    "plugins.alerting.alert_history_retention_period": "7d"
  }
}
```

Si previamente creó monitores y desea detener la creación de índices de alerta diaria, elimine todos los índices del historial de alertas:

```
DELETE .plugins-alerting-alert-history-*
```

Para reducir el recuento de particiones de los índices del historial, cree una plantilla de índice. La siguiente solicitud establece índices de historial para las alertas y administración de estados de índice para una partición y una réplica:

```
PUT _index_template/template-name
{
  "index_patterns": [".opendistro-alerting-alert-history-*"],
  "template": {
    "settings": {
      "number_of_shards": 1,
      "number_of_replicas": 1
    }
  }
}
```

En función de la tolerancia a la pérdida de datos, incluso podría considerar no utilizar ninguna réplica. Para obtener más información acerca de cómo crear y administrar plantillas de índice, consulte Plantillas de índice en la documentación de OpenSearch.

Detección de anomalías en Amazon OpenSearch Service

La detección de anomalías de Amazon OpenSearch Service detecta automáticamente anomalías en los datos de OpenSearch casi en tiempo real mediante el algoritmo de bosque de corte aleatorio (Random Cut Forest, RCF). RCF es un algoritmo de machine learning no supervisado que modela un esquema del flujo de datos entrante. El algoritmo calcula un `anomaly grade` y valor `confidence score` para cada punto de datos entrante. La detección de anomalías utiliza estos valores para diferenciar una anomalía de las variaciones habituales de los datos.

Puede emparejar el complemento de detección de anomalías con el complemento Alertas para recibir una notificación en cuanto se detecte una anomalía.

La detección de anomalías está disponible en dominios que ejecutan cualquier versión de OpenSearch o Elasticsearch 7.4 o posterior. Todos los tipos de instancias admiten la detección de anomalías, excepto `t2.micro` y `t2.small`.

> **Note**
>
> Esta documentación proporciona una breve información general de la detección de anomalías en el contexto de Amazon OpenSearch Service. Para obtener la documentación completa, que incluye los pasos detallados, una referencia de la API, una referencia de toda la configuración disponible y pasos para crear visualizaciones y paneles, consulte Anomaly detection en la documentación de código abierto de OpenSearch.

Requisitos previos

La detección de anomalías posee los siguientes requisitos previos:

- La detección de anomalías requiere OpenSearch o Elasticsearch 7.4 o posterior.
- La detección de anomalías solo admite el control de acceso detallado en las versiones 7.9 y posteriores de Elasticsearch y en todas las versiones de OpenSearch. En las versiones anteriores a Elasticsearch 7.9, solo los usuarios administradores pueden crear, ver y administrar detectores.
- Si el dominio utiliza un control de acceso detallado, los usuarios que no son administradores deben estar mapeados al rol `anomaly_read_access` en OpenSearch Dashboards para ver detectores o al rol `anomaly_full_access` para crear y administrar detectores.

Introducción a la detección de anomalías

Para empezar, elija Anomaly Detection (Detección de anomalías) en OpenSearch Dashboards.

Paso 1: crear un detector

Un detector es una tarea individual de detección de anomalías. Puede crear varios detectores y todos los detectores se pueden ejecutar simultáneamente; cada uno podrá analizar datos de distintos orígenes.

Paso 2: agregar características a su detector

Una característica es el campo del índice que se verifica para ver si hay anomalías. Un detector puede detectar anomalías en una o varias características. Debe elegir una de las siguientes agrupaciones para cada característica: `average()`, `sum()`, `count()`, `min()` o `max()`.

> **Note**
>
> El método de suma `count()` solo está disponible en OpenSearch y Elasticsearch 7.7 o posterior. Para Elasticsearch 7.4, utilice una expresión personalizada como la siguiente:
>
> ```
> {
>   "aggregation_name": {
>     "value_count": {
>       "field": "field_name"
>     }
>   }
> }
> ```

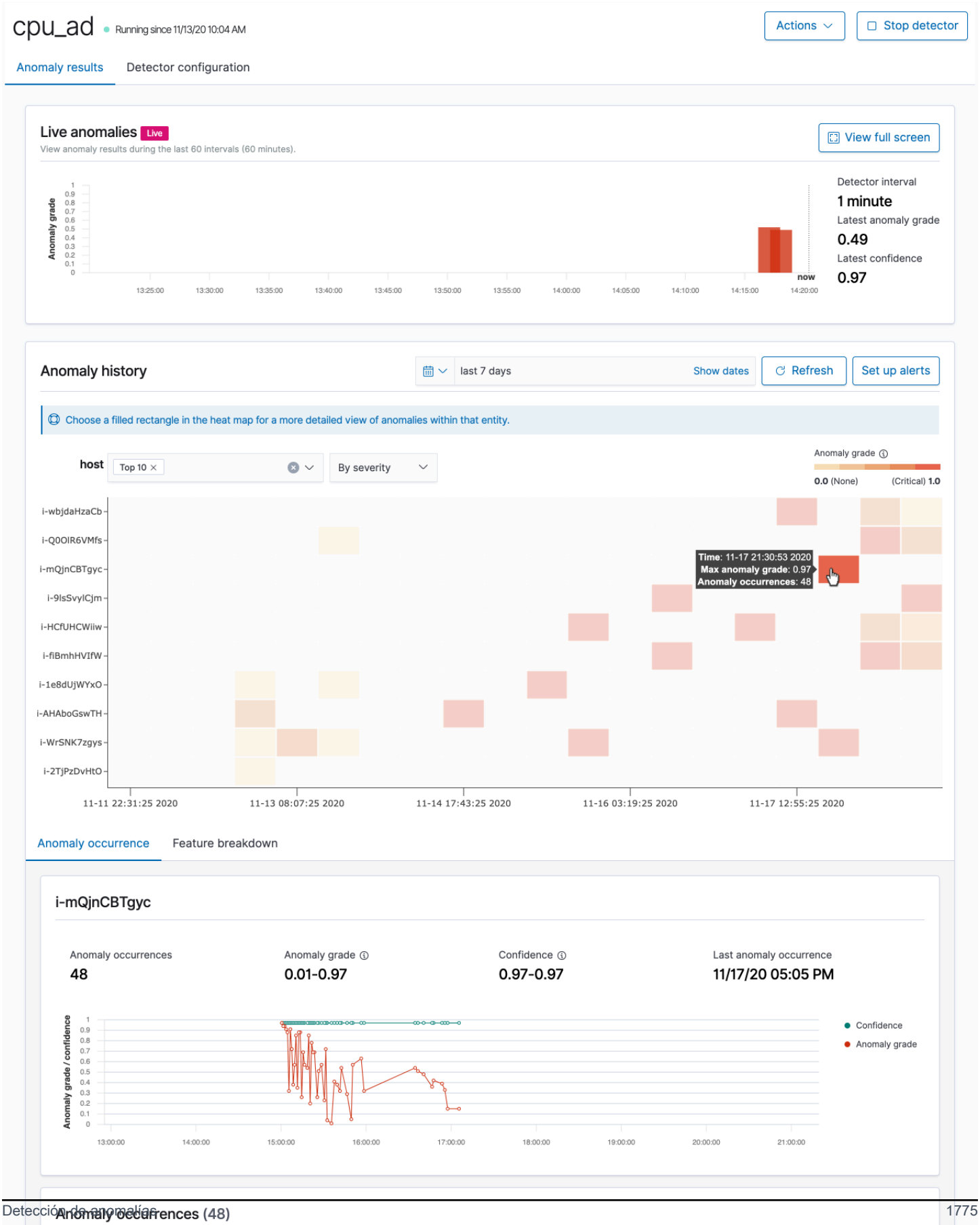
El método de agrupación determina qué constituye una anomalía. Por ejemplo, si elige `min()`, el detector se centra en buscar anomalías basadas en los valores mínimos de la característica. Si elige `average()`, el detector busca anomalías basadas en los valores medios de la característica. Puede agregar un máximo de cinco características por detector.

Puede configurar los siguientes parámetros opcionales (disponibles en Elasticsearch 7.7 y posterior):

- Category field (Categoría): clasifique o corte los datos con una dimensión como dirección IP, ID del producto, código de país, etc.
- Window size (Tamaño de ventana): establezca el número de intervalos de suma del flujo de datos que se tendrán en cuenta en una ventana de detección.

Después de configurar las características, obtenga una previsualización de las muestras de anomalías y ajuste la configuración de las características, si es necesario.

Paso 3: observar los resultados



- Live anomalies (Anomalías activas): muestra los resultados de anomalías activas de los últimos 60 intervalos. Por ejemplo, si el intervalo se establece en 10, muestra los resultados de los últimos 600 minutos. Este gráfico se actualiza cada 30 segundos.
- Anomaly history (Historial de anomalías): traza el grado de anomalía con la medida de confianza correspondiente.
- Feature breakdown (Desglose de características): traza las características según el método de agrupación. Puede variar el rango de fecha y hora del detector.
- Anomaly occurrence (Ocurrencia de anomalías): muestra los valores `Start time`, `End time`, `Data confidence` y `Anomaly grade` de cada anomalía detectada.

  Si establece el campo de categoría, verá un grafico Heat map (Mapa térmico) que correlaciona los resultados de entidades anómalas. Seleccione un rectángulo relleno para ver una vista más detallada de la anomalía.

Paso 4: configurar alertas

Para crear un monitor que envíe notificaciones cuando se detectan anomalías, elija Set up alerts (Configurar alertas). El complemento lo redirige a la página Add monitor (Agregar monitor) donde puede configurar una alerta.

Tutorial: Detectar un uso elevado de la CPU con la detección de anomalías

Este tutorial muestra cómo crear un detector de anomalías en Amazon OpenSearch Service para detectar un uso elevado de la CPU. Utilizará los paneles de OpenSearch para configurar un detector que supervise el uso de la CPU y genere una alerta cuando su uso supere un umbral especificado.

> **Note**
>
> Estos pasos se aplican a la versión más reciente de OpenSearch y pueden diferir ligeramente para las versiones anteriores.

Requisitos previos

- Debe tener un dominio de OpenSearch Service que ejecute Elasticsearch 7.4 o versiones posteriores, o cualquier versión de OpenSearch.

- Debe ingerir archivos de registro de aplicaciones en el clúster que contengan datos de uso de la CPU.

Paso 1: crear un detector

Primero, cree un detector que identifique las anomalías en los datos de uso de la CPU.

1. Abra el menú del panel izquierdo en los paneles de OpenSearch y elija Anomaly Detection (Detección de anomalías) y, después, elija Create detector (Crear detector).
2. Asigne el nombre **`high-cpu-usage`** al detector.
3. Para el origen de datos, elija el índice que contenga los archivos de registro de uso de la CPU en los que desee identificar anomalías.
4. Seleccione el icono Timestamp field (Campo de marca temporal) de los datos. Opcionalmente, puede agregar un filtro de datos. Este filtro de datos analiza únicamente un subconjunto del origen de datos y reduce el ruido de los datos que no son relevantes.
5. Establezca Detector interval (Intervalo de detector) en 2 minutos. Este intervalo define el tiempo (por intervalo de minutos) que tendrá el detector para recopilar los datos.
6. En Window delay (Demora del plazo), agregue una demora de 1 minute (1 minuto). Esta demora agrega tiempo de procesamiento adicional para garantizar que todos los datos dentro del plazo estén presentes.
7. Seleccione Next (Siguiente). En el panel de detección de anomalías, en el nombre del detector, seleccione Configure model (Configurar modelo).
8. En Feature name (Nombre de la característica), ingrese **`max_cpu_usage`**. En Feature state (Estado de la característica), seleccione Enable feature (Habilitar característica).
9. En Find anomalies based on (Encontrar anomalías en función de), elija Field value (Valor de campo).
10. En Aggregation method (Método de agregación), elija **`max()`**.
11. En Field (Campo), seleccione el campo de los datos para ver si hay anomalías. Por ejemplo, podría llamarse `cpu_usage_percentage`.
12. Mantenga todas las demás configuraciones en sus valores predeterminados y elija Next (Siguiente).
13. Ignore la configuración de trabajos del detector y elija Next (Siguiente).
14. En la ventana emergente, elija cuándo desea iniciar el detector (automática o manualmente) y, a continuación, elija Confirm (Confirmar).

Ya que se configuró el detector, después de inicializarse, podrá ver los resultados en tiempo real del uso de la CPU en la sección Real-time results (Resultados en tiempo real) del panel del detector. La sección Live anomalies (Anomalías en vivo) muestra cualquier anomalía que se produce cuando los datos se ingieren en tiempo real.

Paso 2: configurar una alerta

Ya que se creó un detector, cree un monitor que invoque una alerta para enviar un mensaje a Slack cuando se detecte un uso de la CPU que cumpla con las condiciones especificadas en la configuración del detector. Recibirá notificaciones de Slack cuando los datos de uno o más índices cumplan con las condiciones que invocan la alerta.

1. Abra el menú del panel izquierdo en los paneles de OpenSearch y elija Alerting (Alertas) y, a continuación, elija Create monitor (Crear monitor).
2. Proporcione un nombre para el monitor.
3. En Monitor type (Tipo de monitor), elija Per-query monitor (Monitor por consulta). Un monitor por consulta ejecuta una consulta específica y define los activadores.
4. En Monitor defining method (Método de definición de monitor), elija Anomaly detector (Detector de anomalías) y, a continuación, seleccione el detector que ha creado en la sección anterior en el menú desplegable Detector.
5. En Schedule (Programación), elija la frecuencia con la que el monitor recopilará datos y recibirá alertas. Para los fines de este tutorial, establezca la programación para que se ejecute cada 7 minutos.
6. En la sección Triggers (Desencadenadores), elija Add trigger (Agregar desencadenador). En Trigger name (Nombre del desencadenador), ingrese **`High CPU usage`**. En este tutorial, en Severity level (Nivel de gravedad), elija 1, que es el nivel de gravedad más alto.
7. En Anomaly grade threshold (Umbral de clasificación de anomalías), elija IS ABOVE (ES SUPERIOR). En el menú que hay debajo, elija el umbral de clasificación que desea aplicar. En este tutorial, establezca Clasificación de anomalías en 0,7.
8. En Anomaly confidence threshold (Umbral de confianza de anomalías), elija IS ABOVE (ES SUPERIOR). En el menú que hay debajo, ingrese el mismo número que su clasificación de anomalías. En este tutorial, establezca Anomaly confidence threshold (Umbral de confianza de anomalías) en 0,7.
9. En la sección Actions (Acciones), elija Destination (Destino). En el campo Name (Nombre), elija el nombre del destino. En el menú Type (Tipo), elija Slack. En el campo Webhook URL (URL del webhook), ingrese una URL de webhook donde quiera recibir alertas. Para obtener más

información, consulte Sending messages using incoming webhooks (Envío de mensajes mediante webhooks entrantes).

10. Seleccione Create (Crear).

Recursos relacionados

- the section called “Alertas”

- the section called “Detección de anomalías”
- API de detección de anomalías

Amazon Q Developer para Amazon OpenSearch Service

Amazon Q Developer es un asistente impulsado por IA generativa que ayuda a los desarrolladores y profesionales de TI a realizar una variedad de tareas a lo largo del ciclo de vida del desarrollo de software. Amazon Q ayuda a los clientes de AWS a codificar, probar, implementar, solucionar problemas y optimizar sus aplicaciones en ejecución en AWS.

La integración de Amazon Q con Amazon OpenSearch Service ofrece las siguientes funciones generativas:

- the section called “Genere visualizaciones utilizando un lenguaje natural”
- the section called “Ver resúmenes e información sobre alertas”
- the section called “Consulta los resúmenes de los resultados de las consultas generados por Amazon Q en la página Discover”
- the section called “Vea los detectores de anomalías recomendados”
- the section called “Acceda al chat de Amazon Q para preguntas sobre OpenSearch Service”

Para acceder a las características de Amazon Q Developer en OpenSearch Service que son relevantes para su tarea, busque el siguiente ícono en contexto en las páginas Alertas, Descubrir y Crear visualización. Haga clic en el ícono para la petición de asistencia y utilizar las características descritas en esta sección.



Usted también puede chatear directamente con Amazon Q haciendo clic en el ícono en la esquina superior derecha de la consola de OpenSearch Service. El chat de Amazon Q admite respuestas a preguntas sobre las características y funcionalidades de OpenSearch Service.

> **Note**
>
> El chat de Amazon Q no puede acceder a sus datos. Por este motivo, no puede entablar una conversación con el chatbot sobre sus datos.

Para obtener más información sobre Amazon Q, consulte ¿Qué es Amazon Q Developer? en la Guía del usuario de Amazon Q.

Regiones de AWS compatible

El soporte de Amazon Q para OpenSearch Service está disponible en las siguientes Regiones de AWS:

- Este de EE. UU. (Norte de Virginia)
- Oeste de EE. UU. (Oregón)
- Asia-Pacífico (Mumbai)
- Asia-Pacífico (Sídney)
- Asia-Pacífico (Tokio)
- Canadá (centro)
- Europa (Fráncfort)
- Europa (Londres)
- Europa (París)
- América del Sur (São Paulo)

Configuración de Amazon Q para OpenSearch Service

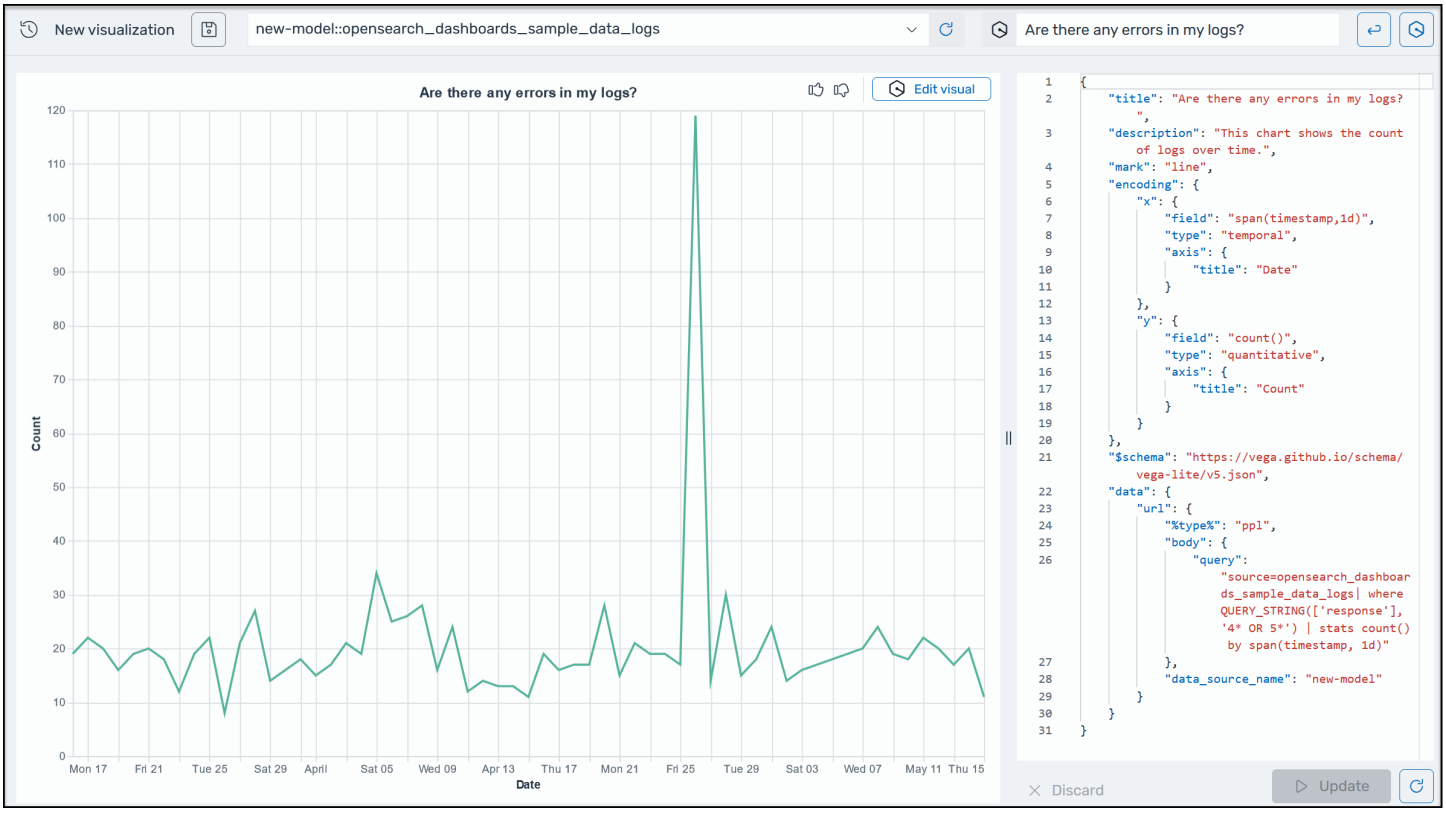
Complete los siguientes pasos para configurar Amazon Q para OpenSearch Service.

1. Verifique haber configurado el control de acceso detallado de OpenSearch Service. Para obtener más información, consulte Control de acceso detallado en Amazon OpenSearch Service.
2. Verifique que su origen de datos esté en OpenSearch versión 2.17 o posterior.
3. Verifique haber seleccionado la casilla Habilitar generación de consulta en lenguaje natural en la sección Inteligencia artificial (IA) y machine learning (ML) durante la creación del dominio o editando la configuración del clúster.

> **Note**
>
> Las características de Amazon Q para Developer OpenSearch están disponibles con el nivel gratuito de Q. Para obtener más información, consulte Comprensión de los niveles de servicio en la Guía del usuario de Amazon Q.

Genere visualizaciones utilizando un lenguaje natural

Para ayudarlo a obtener más información sobre sus datos operativos, Amazon Q Developer para OpenSearch Service permite usar peticiones en lenguaje natural para crear visualizaciones. Puede generar visualizaciones como el siguiente ejemplo usando lenguaje natural desde la página Visualizations o desde la página Discover.



Las visualizaciones pueden agilizar la resolución de problemas al desglosar el análisis de errores según diferentes dimensiones. También pueden ayudarlo a identificar patrones y tendencias, acelerar la toma de decisiones, revelar relaciones y simplificar datos complejos.

Para generar una visualización usando lenguaje natural:

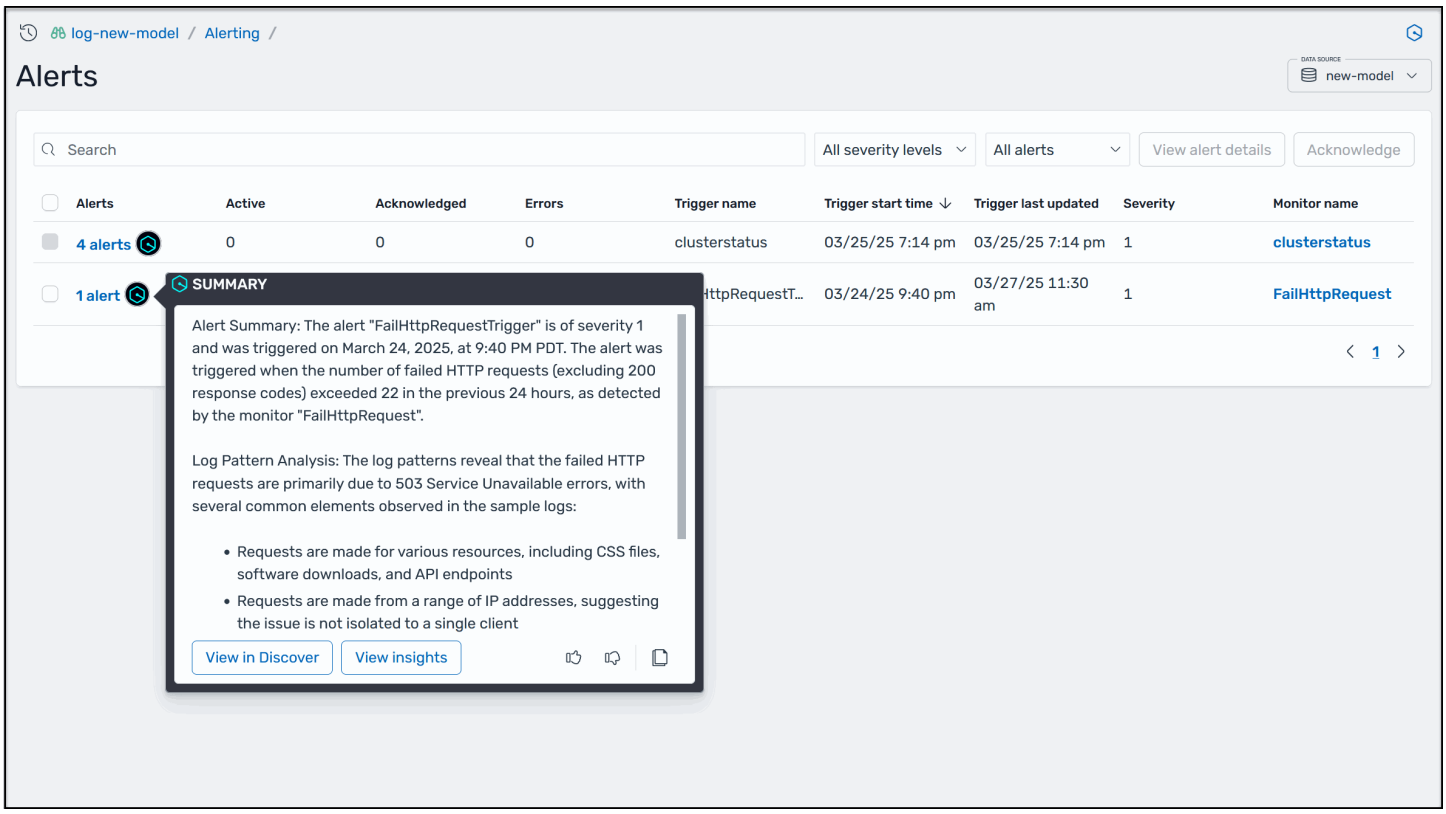
1. Verifique que haya configurado Amazon Q para OpenSearch Service.
2. En el menú principal de OpenSearch Dashboards, elija la página Descubrir y luego seleccione un origen de datos.
3. En el menú Amazon Q, seleccione Generar visualización, como se muestra en la siguiente captura de pantalla.



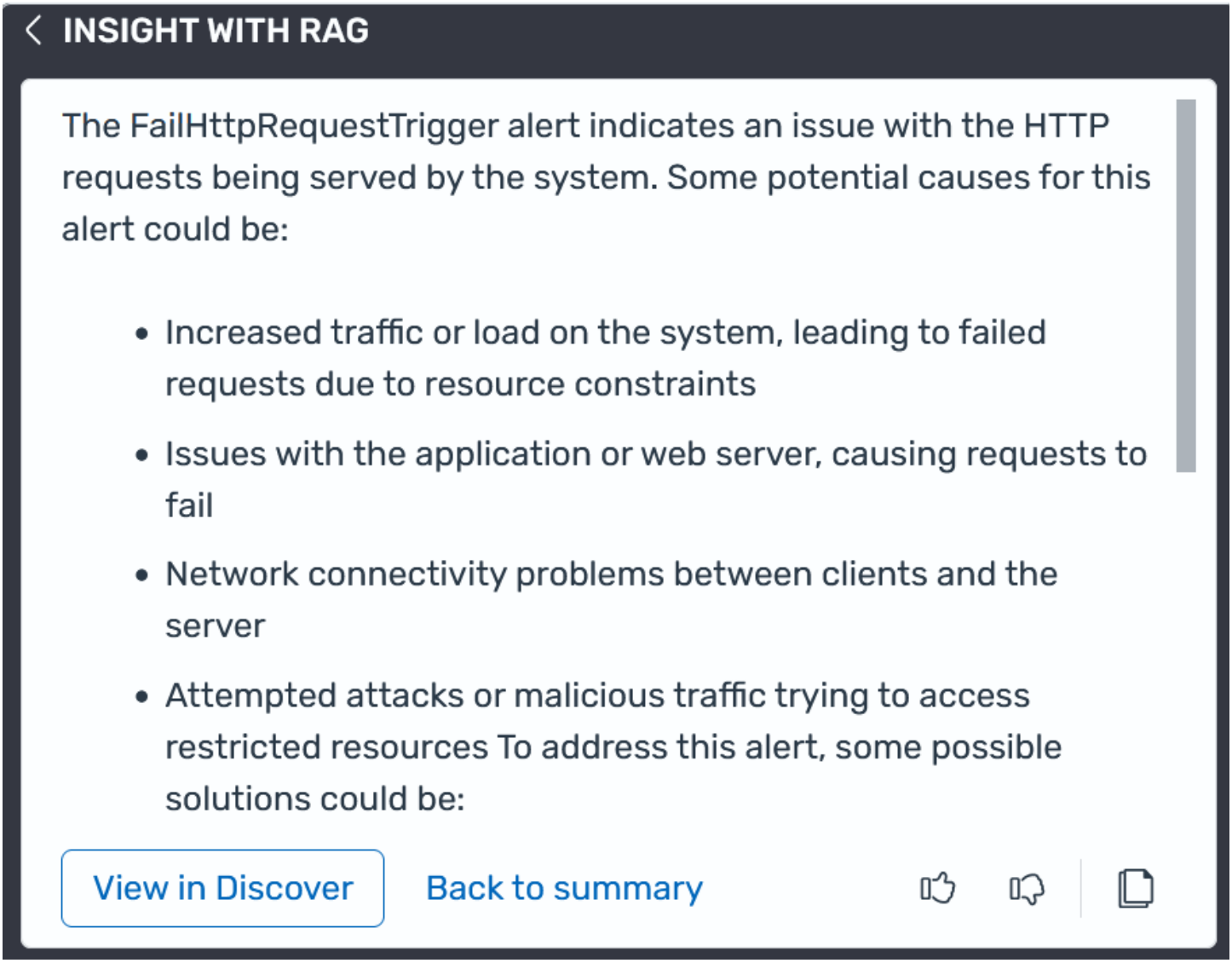
4. Si ingresa una consulta en lenguaje natural en la página Descubrir, Amazon Q copia el contexto al crear una visualización basada en esa consulta. Si todavía no ingresó una consulta en lenguaje natural, en el cuadro de texto de Amazon Q ingrese una petición y luego haga clic en el botón junto al cuadro de texto para ejecutar la consulta. Amazon Q puede tardar unos segundos en crear la visualización.
5. Para actualizar la visualización, seleccione el botón Editar visual en la esquina superior derecha de la visualización. Escriba una nueva petición, por ejemplo: “Cambie esto a un gráfico de líneas” y luego seleccione Aplicar.

Ver resúmenes e información sobre alertas

Puede configurar OpenSearch Service para crear un monitor de alertas cuando los datos de uno o más índices cumplan ciertas condiciones. Para ayudarlo a comprender y solucionar una alerta rápidamente, puede ver un resumen de la alerta haciendo clic en el ícono de Amazon Q Developer junto a una alerta. Un resumen proporciona detalles sobre el problema subyacente que activó la alerta y, cuando está disponible, un análisis adicional para ayudarlo a localizar la causa raíz del problema. La siguiente captura de pantalla muestra un ejemplo de un resumen de alerta creado por Amazon Q.



Si conecta una base de conocimientos para proporcionar contexto adicional sobre su entorno, como se describe más adelante en este tema, Amazon Q crea conocimientos acerca de una alerta. Los conocimientos proporcionan detalles y opciones de solución de problemas para ayudarlo a corregir la causa raíz de una alerta. Para la alerta mostrada anteriormente, Amazon Q también generó los siguientes conocimientos.



> **Note**
>
> Según la naturaleza de la alerta y la información disponible, Amazon Q puede darle la opción de ver los datos de la alerta en la página Descubrir en OpenSearch Dashboards. Si ve el botón Ver en Descubriren la parte inferior de un resumen de alerta de Amazon Q, haga clic en ese botón para abrir el conjunto de datos correspondiente en Descubrir con un filtro activo para los datos de la alerta.

Temas

- Antes de empezar
- Visualización de resúmenes de alertas y conocimientos

Antes de empezar

Complete los siguientes pasos para configurar una base de conocimientos de Amazon Bedrock para que Amazon Q pueda crear conocimientos para las alertas de OpenSearch Service.

Paso 1: Cree el rol de IAM LambdaInvokeOpenSearchMLCommonsRole

Cree un rol nuevo llamado `LambdaInvokeOpenSearchMLCommonsRole` en AWS Identity and Access Management (IAM). OpenSearch Service utiliza este rol para crear un conector de IA en OpenSearch que ayuda a generar conocimientos basados en los artículos de la base de conocimientos configurada. Usted debe asignar este rol `ml_full_access` de OpenSearch Service, como se describe en el paso 2.

Cuando cree el rol nuevo, para Tipo de entidad de confianza, seleccione cuenta de AWS. No necesita especificar una política de permisos. En la página Agregar permisos, elija Siguiente. Para obtener más información sobre cómo crear un rol nuevo, consulte Crear un rol para un servicio de AWS (consola).

Paso 2: Asigne el rol LambdaInvokeOpenSearchMLCommonsRole al rol ml_full_access de OpenSearch Service

Utilice el siguiente procedimiento para asignar el rol `LambdaInvokeOpenSearchMLCommonsRole` al rol `ml_full_access` de OpenSearch Service. Esta asignación también ayuda a que OpenSearch Service cree el conector de IA.

Para asignar el rol de IAM requerido al rol ml_full_access de OpenSearch Service:

1. Abra la página Administración de datos del panel de OpenSearch Service.
2. En Acceso a datos y usuario, elija Roles.
3. Utilice el cuadro de búsqueda para localizar el rol `ml_full_access`.
4. En la página ml_full_access, seleccione la pestaña Usuarios asignados.
5. Elija Asignar usuarios.
6. En el campo Roles de backend, pegue el Nombre de recurso de Amazon (ARN) del rol `LambdaInvokeOpenSearchMLCommonsRole` y luego seleccione Asignar.

Paso 3: Configure una base de conocimientos de OpenSearch Service mediante CloudFormation

Utilice el siguiente procedimiento para configurar una base de conocimientos de OpenSearch Service usando AWS CloudFormation para que Amazon Q pueda generar conocimientos.

Para configurar una base de conocimientos para generar conocimientos:

1. Inicie sesión en la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home en una Región de AWS compatible. Para obtener más información, consulte Regiones de AWS compatible.
2. En el panel de navegación, elija Integraciones.
3. En la sección Plantillas de integración, elija la plantilla Integrar con base de conocimientos mediante Amazon Bedrock. Si no ve esta plantilla, verifique que se encuentre en una Región compatible.
4. En el recuadro Integrar con base de conocimientos mediante Amazon Bedrock, elija Configurar dominio y luego seleccione una de las opciones disponibles. OpenSearch Service abre la plantilla de pila CloudFormation con los campos requeridos previamente completados. La pila CloudFormation admite la integración para dominios públicos y de VPC.
5. Seleccione Creación de pila. Después de que CloudFormation cree los recursos, el servicio muestra el AgentId, el ConnectorId y el ModelId del agente de Amazon Bedrock.

Cuando corresponde, Amazon Q ahora genera conocimientos para las alertas de OpenSearch Service.

Visualización de resúmenes de alertas y conocimientos

Utilice el siguiente procedimiento para ver resúmenes e información sobre alertas en OpenSearch Service.

Visualización de resúmenes de alertas y conocimientos

1. Verifique que haya configurado Amazon Q para OpenSearch Service.
2. Verifique que haya configurado alertas para OpenSearch Service.
3. En el menú principal de OpenSearch Dashboards, elija Creación de alertas y luego Alertas.
4. Seleccione el ícono de Amazon Q junto a una alerta. Amazon Q puede tardar hasta 10 segundos en generar el resumen.

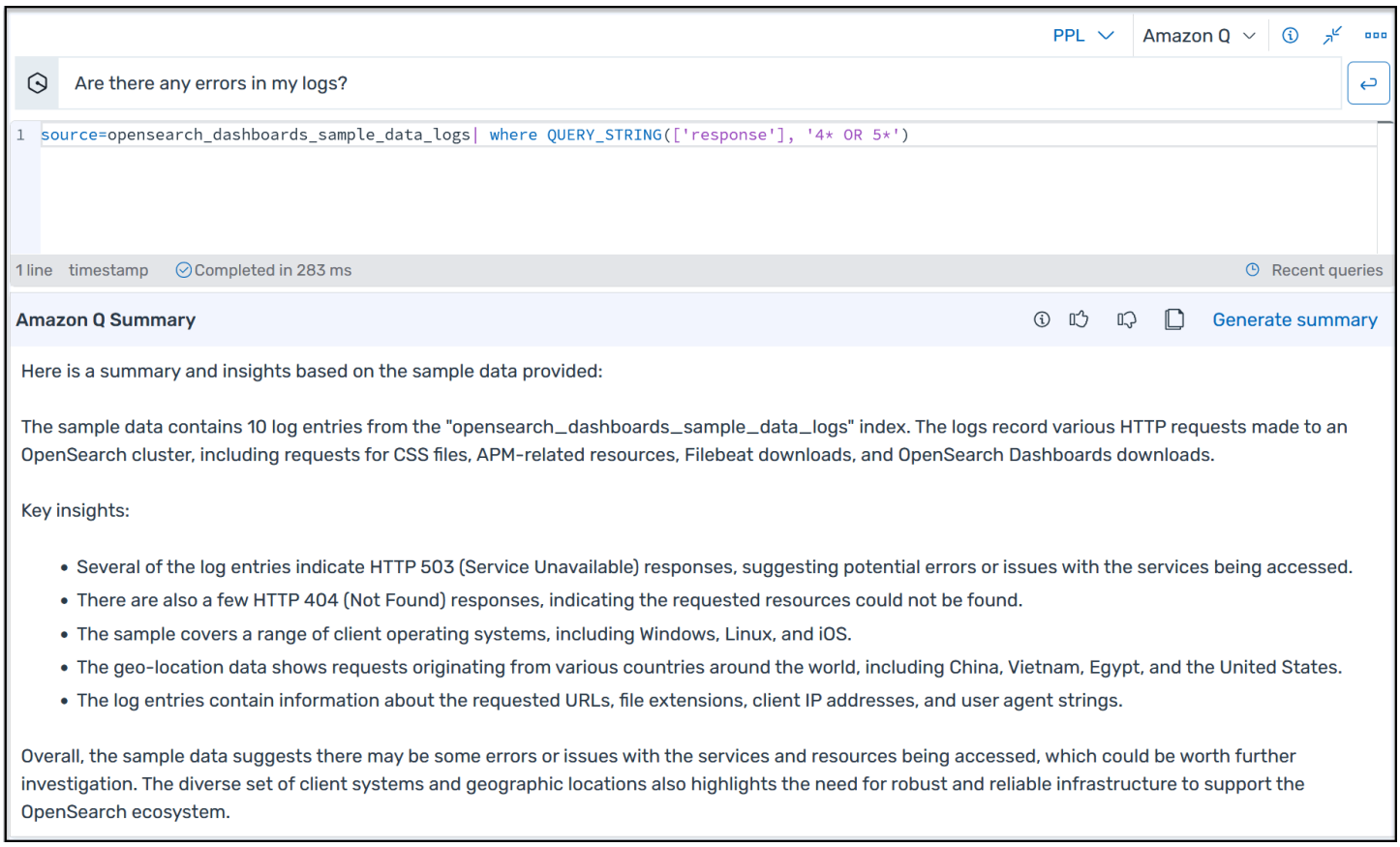
5. Si aparece en el resumen de alertas, elija Ver información para ver más detalles sobre la alerta según su base de conocimiento configurada.
6. Si aparece en el resumen de alertas, elija Ver en Descubrir para ver los datos de la alerta en la página Descubrir de OpenSearch Dashboards.

Consulta los resúmenes de los resultados de las consultas generados por Amazon Q en la página Discover

OpenSearch Service le permite consultar sus datos con peticiones en lenguaje natural mediante el lenguaje de consultas Piped Processing Language (PPL) en la página Descubrir. Por ejemplo, usted puede escribir consultas como las siguientes:

- ¿Hay algún error en mis registros de errores?
- ¿Cuál es el tamaño promedio de las solicitudes por semana?
- ¿Cuántas solicitudes hubo, agrupadas por código de respuesta, la semana pasada?

En respuesta, Amazon Q genera resúmenes en lenguaje natural de los resultados de su consulta basados en los primeros diez registros, como los siguientes:



La combinación de la generación de consultas en lenguaje natural y los resúmenes de consultas puede agregar un nivel adicional de análisis al solucionar una alerta, o brindar una manera más accesible de comprender sus datos sin tener que escribir consultas complejas.

Para ver los resúmenes de resultados de consultas generados por Amazon Q en la página Descubrir

1. Verifique que haya configurado Amazon Q para OpenSearch Service.
2. En el menú principal de OpenSearch Dashboards, elija Descubrir.
3. En la lista desplegable del lenguaje de consultas, elija PPL.
4. En el cuadro de texto de Amazon Q, introduzca una petición y luego haga clic en el botón junto al cuadro de texto para ejecutar la consulta. Amazon Q puede tardar hasta 10 segundos en devolver un resumen. Después de su consulta inicial, usted debe elegir Generar resumen para obtener resúmenes posteriores.

> **Note**
>
> Puede desactivar la generación de resúmenes desde la lista desplegable de Amazon Q en Descubrir.

Vea los detectores de anomalías recomendados

La detección de anomalías de Amazon OpenSearch Service detecta automáticamente anomalías en los datos de OpenSearch casi en tiempo real mediante el algoritmo de bosque de corte aleatorio (Random Cut Forest, RCF). RCF es un algoritmo de machine learning no supervisado que modela un esquema del flujo de datos entrante. El algoritmo calcula un `anomaly grade` y valor `confidence score` para cada punto de datos entrante. La detección de anomalías utiliza estos valores para diferenciar una anomalía de las variaciones habituales de los datos.

Para simplificar el proceso de creación de detectores de anomalías, Amazon Q puede generar detectores sugeridos según el origen de datos que usted seleccione en la página Descubrir. Amazon Q admite detectores de anomalías sugeridos para cualquier idioma.

Para ver los detectores de anomalías recomendados por Amazon Q

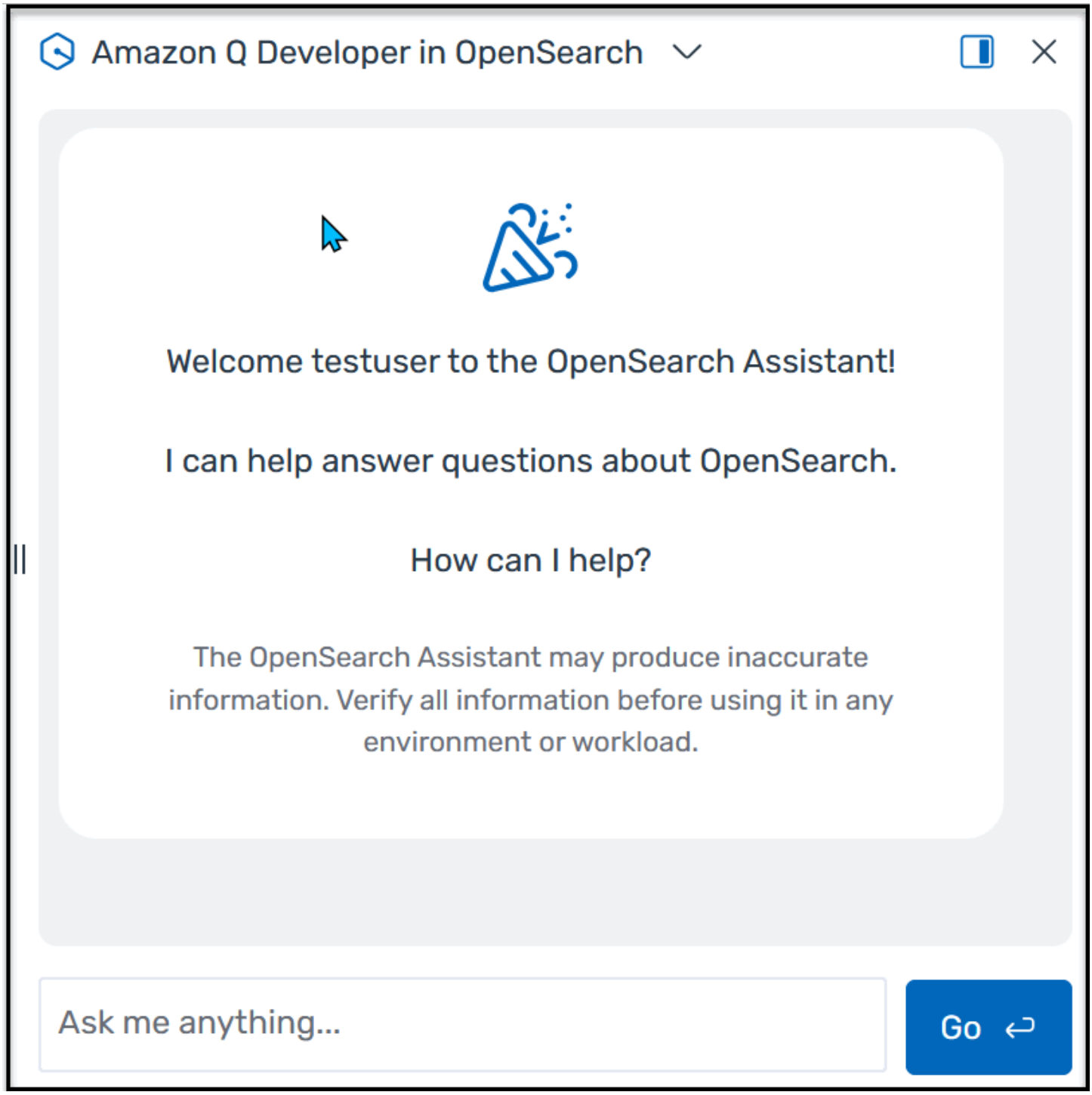
1. Verifique que haya configurado Amazon Q para OpenSearch Service.
2. En el menú principal de OpenSearch Dashboards, elija la página Descubrir y luego seleccione un origen de datos.
3. En el menú de Amazon Q, elija Sugerir detector de anomalías, como se muestra en la siguiente captura de pantalla.



   Amazon Q puede tardar unos segundos en generar las características para el detector.
4. Seleccione Crear detector.

Acceda al chat de Amazon Q para preguntas relacionadas con OpenSearch

Si tiene preguntas sobre Amazon OpenSearch Service, incluidas preguntas conceptuales o procedimentales relacionadas con las características o la funcionalidad de OpenSearch Service, puede pedir información a Amazon Q Developer. El chatbot admite un chat conversacional que conserva el contexto de su conversación. También guarda un historial de sus conversaciones para consultarlo más adelante. Para acceder al chatbot, haga clic en el ícono de Amazon Q en la esquina superior derecha de cualquier página de OpenSearch Service. Escriba una pregunta en el cuadro de texto y haga clic en Ir. Amazon Q puede tardar hasta 30 segundos en generar una respuesta.



Aquí tiene un ejemplo de una respuesta de Amazon Q.

Can you suggest some node management best practices?

Based on your context of being in the OpenSearch Dashboard in the us-west-2 region, here are some node management best practices for Amazon OpenSearch Service: [1]

1. Implement Dedicated Manager Nodes:
    - Use dedicated manager nodes to improve cluster stability. [2]
    - Configure three dedicated manager nodes for optimal domain stability across three Availability Zones.
    - This offloads cluster management tasks from data nodes, enhancing overall performance.
2. Multi-AZ Deployment:
    - Deploy your domain across multiple Availability Zones for increased availability and resilience.
    - Consider using the Multi-AZ with Standby option for 99.99% availability and consistent performance.
3. Proper Sizing:

> **Note**
>
> El chat de Amazon Q no puede acceder a sus datos. Por este motivo, no puede entablar una conversación con el chatbot sobre sus datos.

Machine learning para Amazon OpenSearch Service

ML Commons es un complemento de OpenSearch que proporciona un conjunto de algoritmos de machine learning (ML) comunes a través del transporte y las llamadas de API de REST. Esas llamadas eligen los nodos y los recursos correctos para cada solicitud de ML y supervisan las tareas de ML para garantizar el tiempo de actividad. Esto le permite aprovechar los algoritmos de ML de código abierto existentes y reducir el esfuerzo necesario para desarrollar nuevas características de ML. Para obtener más información sobre el complemento, consulte Machine learning en la documentación de OpenSearch. En este capítulo se explica cómo utilizar el complemento con Amazon OpenSearch Service.

Temas

- Conectores de ML de Amazon OpenSearch Service para Servicios de AWS
- Conectores de ML de Amazon OpenSearch Service para plataformas de terceros
- Uso de CloudFormation para configurar la inferencia remota para la búsqueda semántica
- Configuración de ML Commons no compatible
- Plantillas de Flow Framework de OpenSearch Service

Conectores de ML de Amazon OpenSearch Service para Servicios de AWS

Cuando utilice los conectores de machine learning (ML) de Amazon OpenSearch Service con otro Servicio de AWS, necesitará configurar un rol de IAM para conectar OpenSearch Service a ese servicio de forma segura. Los Servicios de AWS a los que puede configurar un conector incluyen Amazon SageMaker AI y Amazon Bedrock. En este tutorial, explicamos cómo crear un conector de OpenSearch Service a tiempo de ejecución de SageMaker. Para obtener más información sobre los conectores, consulte Conectores compatibles.

Temas

- Requisitos previos
- Cómo crear un conector de OpenSearch Service

Requisitos previos

Para crear un conector, debe tener un punto de conexión de dominio de Amazon Sagemaker AI y un rol de IAM que conceda acceso a OpenSearch Service.

Configuración de un dominio de Amazon SageMaker AI

Consulte Implementar un modelo en Amazon SageMaker AI en la Guía para desarrolladores de Amazon SageMaker AI a fin de implementar su modelo de machine learning. Anote la URL del punto de conexión de su modelo, que necesitará para crear un conector de IA.

Creación de un rol de IAM

Configure un rol de IAM para delegar permisos de tiempo de ejecución de Sagemaker a OpenSearch Service. Para crear un rol nuevo, consulte Creación de roles de IAM (consola) en la Guía del usuario de IAM. Si lo desea, puede utilizar un rol existente siempre que tenga el mismo conjunto de privilegios. Si crea un rol nuevo en lugar de usar un rol administrado por AWS, sustituya `opensearch-sagemaker-role` en este tutorial por el nombre de su propio rol.

1. Adjunte la siguiente política de IAM administrada a su nuevo rol para permitir que OpenSearch Service acceda al punto de conexión de SageMaker AI. Para adjuntar una política a un rol, consulte Adición de permisos de identidad de IAM.

   JSON

   ```
   {
       "Version":"2012-10-17",
       "Statement": [
           {
               "Action": [
                   "sagemaker:InvokeEndpointAsync",
                   "sagemaker:InvokeEndpoint"
               ],
               "Effect": "Allow",
               "Resource": "*"
           }
       ]
   }
   ```

2. Siga las instrucciones de Modificación de una política de confianza de rol para editar la relación de confianza del rol. En la siguiente política, sustituya *`service-principle`* por uno de los siguientes principios de servicio para OpenSearch Service o OpenSearch sin servidor:

Para OpenSearch Service

```
opensearchservice.amazonaws.com
```

Para OpenSearch sin servidor

```
ml.opensearchservice.amazonaws.com
```

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Action": [
                "sts:AssumeRole"
            ],
            "Effect": "Allow",
            "Principal": {
                "Service": [
                    "ml.opensearchservice.amazonaws.com"
                ]
            }
        }
    ]
}
```

Le recomendamos que utilice las claves de condición `aws:SourceAccount` y `aws:SourceArn` para limitar el acceso a un dominio específico. La `SourceAccount` es el ID de Cuenta de AWS que pertenece al propietario del dominio y el `SourceArn` es el ARN del dominio. Por ejemplo, puede agregar el siguiente bloque de condición a la política de confianza:

```
"Condition": {
    "StringEquals": {
        "aws:SourceAccount": "account-id"
    },
    "ArnLike": {
        "aws:SourceArn": "arn:aws:es:region:account-id:domain/domain-name"
    }
}
```

Configuración de permisos

Para crear el conector, necesita permiso para transferir el rol de IAM a OpenSearch Service. También necesita tener acceso a la acción `es:ESHttpPost`. Para conceder estos dos permisos, asocie la siguiente política al rol de IAM cuyas credenciales se utilicen para firmar la solicitud:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::111122223333:role/opensearch-sagemaker-
role"
        },
        {
            "Effect": "Allow",
            "Action": "es:ESHttpPost",
            "Resource": "arn:aws:es:us-east-1:111122223333:domain/domain-name/*"
        }
    ]
}
```

Si su usuario o rol no tiene permisos de `iam:PassRole` para transferir su rol, puede que se produzca un error de autorización cuando intente registrar un repositorio en el siguiente paso.

Asigne el rol de ML en OpenSearch Dashboards (si utiliza el control de acceso detallado)

El control de acceso detallado presenta un paso adicional al configurar un conector. Incluso si utiliza autenticación HTTP básica para todos los demás fines, debe asignar el rol `ml_full_access` al rol de IAM que tenga permisos `iam:PassRole` para transferir `opensearch-sagemaker-role`.

1. Desplácese hasta el complemento de OpenSearch Dashboards para ver su dominio de OpenSearch Service. Puede encontrar el punto de conexión de Dashboards en el panel del dominio de la consola de OpenSearch Service.
2. En el menú principal, seleccione Seguridad, Roles y seleccione el rol ml_full_access.

3. Seleccione Usuarios asignados, Administrar mapeo.
4. En Roles de backend, agregue el ARN del rol que tenga permisos para transferir `opensearch-sagemaker-role`.

```
arn:aws:iam::account-id:role/role-name
```

5. Seleccione Asignar y confirme que el usuario o el rol aparecen en Usuarios asignados.

Cómo crear un conector de OpenSearch Service

Para crear un conector, envíe una solicitud `POST` al punto de conexión de dominio de OpenSearch Service. Puede usar curl, el cliente Python de muestra, Postman u otro método para enviar una solicitud firmada. Tenga en cuenta que no puede usar una solicitud `POST` en la consola de Kibana. La solicitud tiene el siguiente formato:

```
POST domain-endpoint/_plugins/_ml/connectors/_create
{
   "name": "sagemaker: embedding",
   "description": "Test connector for Sagemaker embedding model",
   "version": 1,
   "protocol": "aws_sigv4",
   "credential": {
      "roleArn": "arn:aws:iam::account-id:role/opensearch-sagemaker-role"
   },
   "parameters": {
      "region": "region",
      "service_name": "sagemaker"
   },
   "actions": [
      {
         "action_type": "predict",
         "method": "POST",
         "headers": {
            "content-type": "application/json"
         },
         "url": "https://runtime.sagemaker.region.amazonaws.com/endpoints/endpoint-id/
invocations",
         "request_body": "{ \"inputs\": { \"question\": \"${parameters.question}\",
 \"context\": \"${parameters.context}\" } }"
      }
   ]
```

```
}
```

Si el dominio reside en una nube privada virtual (VPC), la computadora debe estar conectada a la VPC para que la solicitud cree correctamente el conector de IA. El acceso a una VPC depende de la configuración de red, pero generalmente implica conectarse a una VPN o una red corporativa. Para comprobar que tiene acceso al dominio de OpenSearch Service, diríjase a `https://your-vpc-domain.region.es.amazonaws.com` en un navegador web y compruebe que recibe la respuesta JSON predeterminada.

Cliente Python de muestra

El cliente de Python es más simple de automatizar que una solicitud HTTP y tiene una mejor reutilización. Para crear el conector de IA con el cliente Python, guarde el siguiente código de ejemplo en un archivo Python. El cliente necesita los paquetes AWS SDK para Python (Boto3), `requests` y `requests-aws4auth`.

```
import boto3
import requests
from requests_aws4auth import AWS4Auth

host = 'domain-endpoint/'
region = 'region'
service = 'es'
credentials = boto3.Session().get_credentials()
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key, region, service,
 session_token=credentials.token)

# Register repository
path = '_plugins/_ml/connectors/_create'
url = host + path

payload = {
   "name": "sagemaker: embedding",
   "description": "Test connector for Sagemaker embedding model",
   "version": 1,
   "protocol": "aws_sigv4",
   "credential": {
      "roleArn": "arn:aws:iam::account-id:role/opensearch-sagemaker-role"
   },
   "parameters": {
      "region": "region",
      "service_name": "sagemaker"
```

```
   },
   "actions": [
      {
         "action_type": "predict",
         "method": "POST",
         "headers": {
            "content-type": "application/json"
         },
         "url": "https://runtime.sagemaker.region.amazonaws.com/endpoints/endpoint-id/
invocations",
         "request_body": "{ \"inputs\": { \"question\": \"${parameters.question}\",
 \"context\": \"${parameters.context}\" } }"
      }
   ]
}
headers = {"Content-Type": "application/json"}

r = requests.post(url, auth=awsauth, json=payload, headers=headers)
print(r.status_code)
print(r.text)
```

Conectores de ML de Amazon OpenSearch Service para plataformas de terceros

En este tutorial, explicamos cómo crear un conector de OpenSearch Service a Cohere. Para obtener más información sobre los conectores, consulte Conectores compatibles.

Si utiliza un conector de machine learning (ML) de Amazon OpenSearch Service con un modelo remoto externo, debe almacenar sus credenciales de autorización específicas en AWS Secrets Manager. Puede ser una clave de API o una combinación de nombre de usuario y contraseña. Esto significa que también debe crear un rol de IAM que permita a OpenSearch Service acceder a Secrets Manager para leer desde ahí.

Temas

- Requisitos previos
- Cómo crear un conector de OpenSearch Service

Requisitos previos

Para crear un conector para Cohere o cualquier proveedor externo con OpenSearch Service, debe tener un rol de IAM que le dé a OpenSearch Service acceso a AWS Secrets Manager, donde almacena sus credenciales. También debe almacenar sus credenciales en Secrets Manager.

Creación de un rol de IAM

Configure un rol de IAM para delegar permisos de Secrets Manager a OpenSearch Service. También puede usar el rol de `SecretManagerReadWrite` existente. Para crear un rol nuevo, consulte Creación de roles de IAM (consola) en la Guía del usuario de IAM. Si crea un rol nuevo en lugar de usar un rol administrado por AWS, sustituya `opensearch-secretmanager-role` en este tutorial por el nombre de su propio rol.

1. Adjunte la siguiente política de IAM administrada a su nuevo rol para permitir que OpenSearch Service acceda a los valores de Secrets Manager. Para adjuntar una política a un rol, consulte Adición de permisos de identidad de IAM.

   JSON

   ```
   {
       "Version":"2012-10-17",
       "Statement": [
           {
               "Action": [
                   "secretsmanager:GetSecretValue"
               ],
               "Effect": "Allow",
               "Resource": "*"
           }
       ]
   }
   ```

2. Siga las instrucciones de Modificación de una política de confianza de rol para editar la relación de confianza del rol. En la siguiente política, sustituya *`service-principle`* por uno de los siguientes principios de servicio para OpenSearch Service o OpenSearch sin servidor:

   Para OpenSearch Service

   ```
   opensearchservice.amazonaws.com
   ```

Para OpenSearch sin servidor

`ml.opensearchservice.amazonaws.com`

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Action": [
                "sts:AssumeRole"
            ],
            "Effect": "Allow",
            "Principal": {
                "Service": [
                    "service-principle"
                ]
            }
        }
    ]
}
```

Le recomendamos que utilice las claves de condición `aws:SourceAccount` y `aws:SourceArn` para limitar el acceso a un dominio específico. La `SourceAccount` es el ID de Cuenta de AWS que pertenece al propietario del dominio y el `SourceArn` es el ARN del dominio. Por ejemplo, puede agregar el siguiente bloque de condición a la política de confianza:

```
"Condition": {
    "StringEquals": {
        "aws:SourceAccount": "account-id"
    },
    "ArnLike": {
        "aws:SourceArn": "arn:aws:es:region:account-id:domain/domain-name"
    }
}
```

Configuración de permisos

Para crear el conector, necesita permiso para transferir el rol de IAM a OpenSearch Service. También necesita tener acceso a la acción `es:ESHttpPost`. Para conceder estos dos permisos, asocie la siguiente política al rol de IAM cuyas credenciales se utilicen para firmar la solicitud:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "iam:PassRole",
      "Resource": "arn:aws:iam::111122223333:role/opensearch-secretmanager-role"
    },
    {
      "Effect": "Allow",
      "Action": "es:ESHttpPost",
      "Resource": "arn:aws:es:us-east-1:111122223333:domain/domain-name/*"
    }
  ]
}
```

Si su usuario o rol no tiene permisos de `iam:PassRole` para transferir su rol, puede que se produzca un error de autorización cuando intente registrar un repositorio en el siguiente paso.

Configuración de AWS Secrets Manager

Para almacenar sus credenciales de autorización en Secrets Manager, consulte Create an AWS Secrets Manager secret en la Guía del usuario de AWS Secrets Manager.

Después de que Secrets Manager acepta su par clave-valor como secreto, recibirá un ARN con el formato: `arn:aws:secretsmanager:us-west-2:123456789012:secret:MySecret-a1b2c3`. Mantenga un registro de este ARN, a medida que lo usa y su clave cuando cree un conector en el siguiente paso.

Asigne el rol de ML en OpenSearch Dashboards (si utiliza el control de acceso detallado)

El control de acceso detallado presenta un paso adicional al configurar un conector. Incluso si utiliza autenticación HTTP básica para todos los demás fines, debe asignar el rol `ml_full_access` al rol de IAM que tenga permisos `iam:PassRole` para transferir `opensearch-sagemaker-role`.

1. Desplácese hasta el complemento de OpenSearch Dashboards para ver su dominio de OpenSearch Service. Puede encontrar el punto de conexión de Dashboards en el panel del dominio de la consola de OpenSearch Service.
2. En el menú principal, seleccione Seguridad, Roles y seleccione el rol ml_full_access.
3. Seleccione Usuarios asignados, Administrar mapeo.
4. En Roles de backend, agregue el ARN del rol que tenga permisos para transferir `opensearch-sagemaker-role`.

   ```
   arn:aws:iam::account-id:role/role-name
   ```

5. Seleccione Asignar y confirme que el usuario o el rol aparecen en Usuarios asignados.

Cómo crear un conector de OpenSearch Service

Para crear un conector, envíe una solicitud `POST` al punto de conexión de dominio de OpenSearch Service. Puede usar curl, el cliente Python de muestra, Postman u otro método para enviar una solicitud firmada. Tenga en cuenta que no puede usar una solicitud `POST` en la consola de Kibana. La solicitud tiene el siguiente formato:

```
POST domain-endpoint/_plugins/_ml/connectors/_create
{
    "name": "Cohere Connector: embedding",
    "description": "The connector to cohere embedding model",
    "version": 1,
    "protocol": "http",
    "credential": {
        "secretArn": "arn:aws:secretsmanager:region:account-id:secret:cohere-key-id",
        "roleArn": "arn:aws:iam::account-id:role/opensearch-secretmanager-role"
    },
    "actions": [
        {
            "action_type": "predict",
```

```
            "method": "POST",
            "url": "https://api.cohere.ai/v1/embed",
            "headers": {
                "Authorization": "Bearer ${credential.secretArn.cohere-key-used-in-
secrets-manager}"
            },
            "request_body": "{ \"texts\": ${parameters.texts}, \"truncate\": \"END\" }"
        }
    ]
}
```

El cuerpo de esta solicitud es diferente de una solicitud de conector de código abierto en dos aspectos. Dentro del campo `credential`, se transfiere el ARN del rol de IAM que permite a OpenSearch Service leer desde Secrets Manager, junto con el ARN del secreto qué. En el campo `headers`, se hace referencia al secreto mediante la clave secreta y al hecho de que proviene de un ARN.

Si el dominio reside en una nube privada virtual (VPC), la computadora debe estar conectada a la VPC para que la solicitud cree correctamente el conector de IA. El acceso a una VPC depende de la configuración de red, pero generalmente implica conectarse a una VPN o una red corporativa. Para comprobar que tiene acceso al dominio de OpenSearch Service, diríjase a `https://`*`your-vpc-domain`*`.`*`region`*`.es.amazonaws.com` en un navegador web y compruebe que recibe la respuesta JSON predeterminada.

Cliente Python de muestra

El cliente de Python es más simple de automatizar que una solicitud HTTP y tiene una mejor reutilización. Para crear el conector de IA con el cliente Python, guarde el siguiente código de ejemplo en un archivo Python. El cliente necesita los paquetes AWS SDK para Python (Boto3), `requests` y `requests-aws4auth`.

```
import boto3
import requests
from requests_aws4auth import AWS4Auth

host = 'domain-endpoint/'
region = 'region'
service = 'es'
credentials = boto3.Session().get_credentials()
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key, region, service,
 session_token=credentials.token)
```

```
path = '_plugins/_ml/connectors/_create'
url = host + path

payload = {
    "name": "Cohere Connector: embedding",
    "description": "The connector to cohere embedding model",
    "version": 1,
    "protocol": "http",
    "credential": {
        "secretArn": "arn:aws:secretsmanager:region:account-id:secret:cohere-key-id",
        "roleArn": "arn:aws:iam::account-id:role/opensearch-secretmanager-role"
    },
    "actions": [
        {
            "action_type": "predict",
            "method": "POST",
            "url": "https://api.cohere.ai/v1/embed",
            "headers": {
                "Authorization": "Bearer ${credential.secretArn.cohere-key-used-in-
secrets-manager}"
            },
            "request_body": "{ \"texts\": ${parameters.texts}, \"truncate\": \"END\" }"
        }
    ]
}

headers = {"Content-Type": "application/json"}

r = requests.post(url, auth=awsauth, json=payload, headers=headers)
print(r.status_code)
print(r.text)
```

Uso de CloudFormation para configurar la inferencia remota para la búsqueda semántica

A partir de la versión 2.9 de OpenSearch, puede utilizar la inferencia remota con la búsqueda semántica para alojar sus propios modelos de machine learning (ML). La inferencia remota utiliza el complemento ML Commons.

Con la inferencia remota, puede alojar las inferencias de sus modelos de forma remota en servicios de ML, como Amazon SageMaker AI y Amazon BedRock y conectarlas a Amazon OpenSearch Service mediante conectores de ML.

Para facilitar la configuración de la inferencia remota, Amazon OpenSearch Service proporciona una plantilla AWS CloudFormation en la consola. CloudFormation es un Servicio de AWS que le permite aprovisionar y administrar recursos de AWS y de terceros al tratar la infraestructura como código.

La plantilla de CloudFormation de OpenSearch automatiza el proceso de aprovisionamiento de modelos para que pueda crear fácilmente un modelo en su dominio de OpenSearch Service y, a continuación, utilizar el ID del modelo para incorporar datos y ejecutar consultas de búsqueda neuronal.

Cuando utilice codificadores dispersos neuronales con la versión 2.12 y versiones posteriores de OpenSearch Service, le recomendamos que utilice el modelo de tokenizador de forma local en lugar de implementarlo de forma remota. Para obtener más información, consulte Sparse encoding models en la documentación de OpenSearch.

Temas

- Plantillas CloudFormation disponibles
- Requisitos previos
- Plantillas de Amazon Bedrock
- Plantillas de integración de servidores MCP
- Plantillas Amazon SageMaker
- Inferencia remota para plantillas de resaltado semántico

Plantillas CloudFormation disponibles

Están disponibles las siguientes plantillas AWS CloudFormation de machine learning para su uso:

Plantillas de Amazon Bedrock

Integración de Amazon Titan Text Embeddings

> Se conecta a los modelos de ML alojados de Amazon Bedrock, elimina la necesidad de implementar modelos independientes y utiliza puntos de conexión de Amazon Bedrock predeterminados. Para obtener más información, consulte Amazon Titan Text Embeddings en la Guía del usuario de Amazon Bedrock.

Integración Cohere Embed

Proporciona acceso a los modelos Cohere Embed y está optimizado para flujos de trabajo específicos de procesamiento de texto. Para obtener más información, consulte Embed en el sitio web de documentos de Cohere.

Amazon Titan Multimodal Embeddings

Admite la incrustación de texto e imágenes y permite funciones de búsqueda multimodal. Para obtener más información, consulte Amazon Titan Multimodal Embeddings en la Guía del usuario de Amazon Bedrock.

Plantillas de integración de servidores MCP

Integración de servidores MCP

Implementa un Tiempo de ejecución de Amazon Bedrock AgentCore, proporciona un punto de conexión de agente, gestiona la autenticación entrante y saliente y admite autenticación de salida para la autenticación empresarial.

Plantillas Amazon SageMaker

Integración con modelos de incrustación de texto mediante Amazon SageMaker

Implementa modelos de incrustación de texto en Amazon SageMaker Runtime, crea roles de IAM para el acceso a los artefactos de los modelos y establece conectores de ML para la búsqueda semántica.

Integración con codificadores dispersos mediante SageMaker

Configura modelos de codificación dispersos para la búsqueda neuronal, crea funciones AWS Lambda para la administración de conectores y devuelve los ID de los modelos para su uso inmediato.

Requisitos previos

Para utilizar una plantilla de CloudFormation con OpenSearch Service, se deben completar los siguientes requisitos previos.

Configurar un dominio de OpenSearch Service

Antes de poder usar una plantilla de CloudFormation, debe configurar un dominio de Amazon OpenSearch Service con la versión 2.9 o posterior y activar un control de acceso detallado. Cree un rol de backend de OpenSearch Service para dar permiso al complemento de ML Commons para que cree el conector por usted.

La plantilla de CloudFormation crea un rol de IAM de Lambda por usted con el nombre predeterminado `LambdaInvokeOpenSearchMLCommonsRole`, que puede anular si desea elegir un nombre diferente. Una vez que la plantilla crea este rol de IAM, debe conceder permiso a la función de Lambda para llamar a su dominio de OpenSearch Service. Para ello, asigne el rol denominado `ml_full_access` a su rol de backend de OpenSearch Service siguiendo estos pasos:

1. Desplácese hasta el complemento de OpenSearch Dashboards para ver su dominio de OpenSearch Service. Puede encontrar el punto de conexión de Dashboards en el panel del dominio de la consola de OpenSearch Service.
2. En el menú principal, seleccione Seguridad, Roles y seleccione el rol ml_full_access.
3. Seleccione Usuarios asignados, Administrar mapeo.
4. En Roles de backend, agregue el ARN del rol de Lambda que necesita permiso para llamar a su dominio.

   ```
   arn:aws:iam::account-id:role/role-name
   ```

5. Seleccione Asignar y confirme que el usuario o el rol aparecen en Usuarios asignados.

Una vez que haya asignado el rol, vaya a la configuración de seguridad de su dominio y añada el rol de IAM de Lambda a su política de acceso de OpenSearch Service.

Habilite los permisos en su Cuenta de AWS

Su Cuenta de AWS debe tener permiso para acceder a CloudFormation y Lambda, así como a cualquier Servicio de AWS que elija para su plantilla, ya sea tiempo de ejecución de Sagemaker o Amazon BedRock.

Si utiliza Amazon Bedrock, también debe registrar su modelo. Consulte Acceso al modelo en la Guía del usuario de Amazon Bedrock para registrar su modelo.

Si utiliza su propio bucket de Amazon S3 para proporcionar artefactos del modelo, debe añadir el rol de IAM de CloudFormation a su política de acceso de S3. Para más información, consulte Adición y eliminación de permisos de identidad de IAM en la Guía del usuario de IAM.

Plantillas de Amazon Bedrock

Al igual que la plantillas de CloudFormation de Amazon SageMaker, la plantilla de CloudFormation de Amazon Bedrock proporciona los recursos de AWS necesarios para crear conectores entre OpenSearch Service y Amazon Bedrock.

En primer lugar, la plantilla crea un rol de IAM que permite a la futura función de Lambda acceder a su dominio de OpenSearch. A continuación, la plantilla crea la función de Lambda, que hace que el dominio cree un conector mediante el complemento ML Commons. Una vez que OpenSearch Service crea el conector, habrá finalizado la configuración de la inferencia remota y podrá ejecutar búsquedas semánticas mediante las operaciones de la API de Amazon Bedrock.

> **Note**
>
> Tenga en cuenta que, dado que Amazon Bedrock aloja sus propios modelos de ML, no necesita implementar un modelo en tiempo de ejecución de Sagemaker. En su lugar, la plantilla utiliza un punto de conexión predeterminado para Amazon Bedrock y omite los pasos de aprovisionamiento del punto de conexión.

Uso de la plantilla de CloudFormation de Amazon Bedrock

1. Abra la consola de Amazon OpenSearch Service.
2. En el panel de navegación de la izquierda, seleccione Integraciones.
3. En Integrar con el modelo de Amazon Titan Text Embeddings a través de Amazon Bedrock, elija Configurar dominio, Configurar dominio público.
4. Siga la petición para configurar el modelo.

> **Note**
>
> OpenSearch Service también proporciona una plantilla independiente para configurar el dominio de VPC de Amazon. Si usa esta plantilla, debe proporcionar el ID de VPC de Amazon para la función de Lambda.

Además, OpenSearch Service proporciona las siguientes plantillas de Amazon Bedrock para conectarse al modelo de Cohere y al modelo de Amazon Titan Multimodal Embeddings:

- `Integration with Cohere Embed through Amazon Bedrock`
- `Integrate with Amazon Bedrock Titan Multi-modal`

Plantillas de integración de servidores MCP

Con las plantillas de servidor del protocolo de contexto para modelos (MCP), puede implementar un servidor MCP alojado en OpenSearch en Amazon Bedrock AgentCore, lo que reduce la complejidad de la integración entre los agentes de IA y las herramientas de OpenSearch. Para más información, consulte ¿Qué es Amazon Bedrock AgentCore?.

Características de la plantilla

Esta plantilla incluye las siguientes características clave para implementar y administrar su servidor MCP.

Implementación de un servidor MCP administrado

Implementa opensearch-mcp-server-py con tiempo de ejecución de Amazon Bedrock AgentCore y proporciona un punto de conexión de agente que envía las solicitudes por proxy al servidor MCP subyacente. Para obtener más información, consulte opensearch-mcp-server-py en GitHub.

Seguridad y autenticación

Gestiona tanto la autenticación entrante (de los usuarios al servidor MCP) como la autenticación saliente (del servidor MCP a OpenSearch) y admite OAuth para la autenticación empresarial.

> **Note**
>
> La plantilla del servidor MCP solo está disponible en las siguientes Regiones de AWS:
>
> - Este de EE. UU. (Norte de Virginia)
> - Oeste de EE. UU. (Oregón)
> - Europa (Fráncfort)
> - Asia-Pacífico (Sídney)

Para utilizar la plantilla de servidor MCP

Siga estos pasos para implementar la plantilla de servidor MCP y conectarla a su dominio de OpenSearch.

1. Abra la consola de Amazon OpenSearch Service.
2. En el panel de navegación de la izquierda, seleccione Integraciones.
3. Localice la plantilla de integración del servidor MCP.
4. Elija Configurar dominio. A continuación, introduzca el punto de conexión de su dominio de OpenSearch.

La plantilla crea un tiempo de ejecución de AgentCore y los siguientes componentes, si no se especifican los parámetros opcionales correspondientes:

- Un repositorio de Amazon ECR.
- O bien, utilice un grupo de usuarios de Amazon Cognito como autorizador OAuth.
- Una función de ejecución utilizada por el Tiempo de ejecución de AgentCore

Después de completar este procedimiento, debe seguir estos pasos posteriores a la creación:

1. Para Amazon OpenSearch Service: asigne su ARN de rol de ejecución a un rol de backend de OpenSearch para controlar el acceso a su dominio.

   Para Amazon OpenSearch sin servidor: cree una política de acceso a los datos que permita que su rol de ejecución acceda a su colección.
2. Obtenga un token de acceso de OAuth de su autorizador. A continuación, utilice este token para acceder al servidor MCP en la URL que aparece en la salida de la pila de CloudFormation.

Para obtener más información, consulte Acciones de política de OpenSearch sin servidor.

Integración con agentes de IA

Tras la implementación, puede integrar el servidor MCP con cualquier agente compatible con MCP. Para obtener más información, consulte Invocar a su servidor MCP desplegado en la Guía para desarrolladores de Amazon Bedrock.

Integración de desarrolladores

Puede agregar el punto de conexión del servidor MCP a la configuración de su agente. También puede usarlo con la CLI para Amazon Q Developer, agentes personalizados u otros agentes compatibles con MCP.

Implementación empresarial

Los agentes alojados en Central pueden conectarse a varios servicios con OpenSearch como un componente. Este agente es compatible con los sistemas de autenticación OAuth y empresariales y se escala para admitir varios usuarios y casos de uso.

Ejemplo de uso del marco Strands Agents

```
import os
import requests
from strands import Agent
from strands.tools.mcp import MCPClient
from mcp.client.streamable_http import streamablehttp_client

def get_bearer_token(discovery_url: str, client_id: str, client_secret: str):
    response = requests.get(discovery_url)
    discovery_data = response.json()
    token_endpoint = discovery_data['token_endpoint']

    data = {
        'grant_type': 'client_credentials',
        'client_id': client_id,
        'client_secret': client_secret
    }
    headers = {
        'Content-Type': 'application/x-www-form-urlencoded'
    }

    response = requests.post(token_endpoint, data=data, headers=headers)
    token_data = response.json()
    return token_data['access_token']

if __name__ == "__main__":
    discovery_url = os.environ["DISCOVERY_URL"]
    client_id = os.environ["CLIENT_ID"]
    client_secret = os.environ["CLIENT_SECRET"]
    mcp_url = os.environ["MCP_URL"]
```

```
bearer_token = get_bearer_token(discovery_url, client_id, client_secret)

opensearch_mcp_client = MCPClient(lambda: streamablehttp_client(mcp_url, {
    "authorization": f"Bearer {bearer_token}",
    "Content-Type": "application/json"
}))

with opensearch_mcp_client:
    tools = opensearch_mcp_client.list_tools_sync()
    agent = Agent(tools=tools)
    agent("list indices")
```

Para obtener más información, consulte Alojamiento del servidor MCP de OpenSearch con Amazon Bedrock AgentCore en el sitio web de OpenSearch.

Plantillas Amazon SageMaker

La plantilla de CloudFormation de Amazon SageMaker define varios recursos de AWS para configurar el complemento neuronal y la búsqueda semántica.

En primer lugar, utilice la plantilla Integración con modelos de incrustación de texto a través de Amazon SageMaker para implementar un modelo de incrustación de texto en tiempo de ejecución de SageMaker como servidor. Si no proporciona un punto de conexión del modelo, CloudFormation crea un rol de IAM que permite que el tiempo de ejecución de SageMaker descargue artefactos de modelos de Amazon S3 y los implemente en el servidor. Si proporciona un punto de conexión, CloudFormation crea un rol de IAM que permite que la función de Lambda acceda al dominio de OpenSearch Service o, si el rol ya existe, actualiza y reutiliza el rol. El punto de conexión sirve al modelo remoto que se utiliza para el conector de ML con el complemento ML Commons.

A continuación, utilice la plantilla Integración con codificadores dispersos a través de Amazon SageMaker para crear una función de Lambda con la que el dominio configure conectores de inferencia remota. Una vez creado el conector en OpenSearch Service, la inferencia remota puede ejecutar una búsqueda semántica mediante el modelo remoto de tiempo de ejecución de SageMaker. La plantilla le devuelve el ID del modelo de su dominio para que pueda empezar la búsqueda.

Uso de las plantillas de CloudFormation de Amazon SageMaker AI

1. Abra la consola de Amazon OpenSearch Service.

2. En el panel de navegación de la izquierda, seleccione Integraciones.
3. En cada una de las plantillas de Amazon SageMaker AI, elija Configurar dominio y Configurar dominio público.
4. Siga la petición de la consola de CloudFormation para aprovisionar la pila y configurar un modelo.

> **Note**
>
> OpenSearch Service también proporciona una plantilla independiente para configurar el dominio de VPC. Si usa esta plantilla, debe proporcionar el ID de VPC para la función de Lambda.

Inferencia remota para plantillas de resaltado semántico

El resaltado semántico es una característica de búsqueda avanzada que mejora la relevancia de los resultados al analizar el significado y el contexto de las consultas en lugar de basarse únicamente en las coincidencias exactas de las palabras clave. Esta función utiliza modelos de machine learning para evaluar la similitud semántica entre las consultas de búsqueda y el contenido del documento, identificando y resaltando las frases o pasajes más relevantes desde el punto de vista contextual de los documentos. A diferencia de los métodos de resaltado tradicionales, que se centran en las coincidencias exactas de los términos, el resaltado semántico aprovecha los modelos de IA para evaluar cada oración utilizando información contextual tanto de la consulta como del texto circundante, lo que le permite mostrar información pertinente incluso cuando los términos de búsqueda exactos no estén presentes en los pasajes resaltados. Este enfoque es particularmente valioso para las implementaciones de búsqueda impulsadas por la IA, en las que los usuarios dan prioridad al significado semántico por encima de la coincidencia literal de palabras, lo que permite a los administradores de búsquedas ofrecer experiencias de búsqueda más inteligentes y contextualmente conscientes que destacan las extensiones de contenido significativas en lugar de solo las apariciones de palabras clave. Para obtener más información, consulte Uso del resaltado semántico.

Utilice el siguiente procedimiento para abrir y ejecutar una plantilla CloudFormation que configure automáticamente los modelos de Amazon SageMaker para el resaltado semántico.

Uso de la plantilla de CloudFormation de resaltado semántico

1. Abra la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. En el panel de navegación izquierdo, seleccione Integraciones.
3. En Habilitar el resaltado semántico mediante la integración con Amazon Sagemaker, elija Configurar dominio y Configurar dominio público.
4. Siga la petición para configurar el modelo.

> **Note**
>
> OpenSearch Service también proporciona una plantilla independiente para configurar el dominio de VPC. Si usa esta plantilla, debe proporcionar el ID de VPC para la función de Lambda.

Configuración de ML Commons no compatible

Amazon OpenSearch Service no admite el uso de la siguiente configuración de ML Commons:

- `plugins.ml_commons.allow_registering_model_via_url`
- `plugins.ml_commons.allow_registering_model_via_local_file`

> **Important**
>
> En los clústeres de producción, no desactive la configuración del clúster `plugins.ml_commons.only_run_on_ml_node` (no la configure en `false`). La opción de deshabilitar esta protección es para facilitar el desarrollo, pero los clústeres de producción deberían utilizar los conectores. Para obtener más información, consulte the section called “Conectores para Servicios de AWS”.

Para obtener más información sobre la configuración de ML Commons, consulte ML Commons cluster settings.

Plantillas de Flow Framework de OpenSearch Service

Las plantillas de Flow Framework de Amazon OpenSearch Service le permiten automatizar las tareas complejas de configuración y preprocesamiento de OpenSearch Service al proporcionar plantillas para casos de uso comunes. Por ejemplo, puede utilizar plantillas de Flow Framework para automatizar las tareas de configuración de machine learning. Las plantillas de Flow Framework de Amazon OpenSearch Service proporcionan una descripción compacta del proceso de configuración en un documento JSON o YAML. Estas plantillas describen las configuraciones de flujos de trabajo automatizadas para el chat conversacional o la generación de consultas, los conectores de IA, las herramientas, los agentes y otros componentes que preparan OpenSearch Service para su uso de backend en modelos generativos.

Las plantillas de Flow Framework de Amazon OpenSearch Service se pueden personalizar para que se ajuste a sus necesidades específicas. Para ver un ejemplo de una plantilla de Flow Framework personalizada, consulte flow-framework. Para ver las plantillas proporcionadas por OpenSearch Service, consulte workflow-templates. Para obtener una documentación completa, que incluye los pasos detallados, una referencia de la API y una referencia de toda la configuración disponible, consulte Automating configuration en la documentación de código abierto de OpenSearch.

> **Note**
>
> Flow-framework no admite el filtrado de roles de backend para OpenSearch Service 2.17.

Creación de conectores de ML en OpenSearch Service

Las plantillas de Flow Framework de Amazon OpenSearch Service le permiten configurar e instalar conectores de ML mediante la API de creación de conectores que se ofrece en ml-commons. Puede usar conectores de ML para conectar OpenSearch Service a otros servicios de AWS o plataformas de terceros. Para obtener más información al respecto, consulte Creating connectors for third-party ML platforms. La API de Flow Framework de Amazon OpenSearch Service le permite automatizar las tareas de configuración y preprocesamiento de OpenSearch Service y se puede utilizar para crear conectores de ML.

Antes de poder crear un conector en OpenSearch Service, debe hacer lo siguiente:

- Cree un dominio de Amazon SageMaker AI.
- Crear un rol de IAM.

- Configure el permiso para transferir roles.
- Asigne los roles de flow-framework y ml-commons en los paneles de OpenSearch.

Para obtener más información sobre cómo configurar conectores de ML para servicios de AWS, consulte Amazon OpenSearch Service ML connectors for AWS services. Para obtener más información sobre el uso de los conectores de ML de OpenSearch Service con plataformas de terceros, consulte Amazon OpenSearch Service ML connectors for third-party platforms.

Creación de un conector mediante un servicio flow-framework

Para crear una plantilla de flow-framework con un conector, tendrá que enviar una solicitud `POST` al punto de conexión de su dominio de OpenSearch Service. Puede usar cURL, un cliente Python de muestra, Postman u otro método para enviar una solicitud firmada. La solicitud `POST` tiene el siguiente formato:

```
POST /_plugins/_flow_framework/workflow
{
  "name": "Deploy Claude Model",
  "description": "Deploy a model using a connector to Claude",
  "use_case": "PROVISION",
  "version": {
    "template": "1.0.0",
    "compatibility": [
      "2.12.0",
      "3.0.0"
    ]
  },
  "workflows": {
    "provision": {
      "nodes": [
        {
          "id": "create_claude_connector",
          "type": "create_connector",
          "user_inputs": {
            "name": "Claude Instant Runtime Connector",
            "version": "1",
            "protocol": "aws_sigv4",
            "description": "The connector to Bedrock service for Claude model",
            "actions": [
              {
                "headers": {
```

```
                  "x-amz-content-sha256": "required",
                  "content-type": "application/json"
               },
               "method": "POST",
               "request_body": "{ \"prompt\":\"${parameters.prompt}\",
 \"max_tokens_to_sample\":${parameters.max_tokens_to_sample},
 \"temperature\":${parameters.temperature},  \"anthropic_version\":
\"${parameters.anthropic_version}\" }",
               "action_type": "predict",
               "url": "https://bedrock-runtime.us-west-2.amazonaws.com/model/
anthropic.claude-instant-v1/invoke"
             }
          ],
          "credential": {
              "roleArn": "arn:aws:iam::account-id:role/opensearch-secretmanager-
role"
           },
          "parameters": {
            "endpoint": "bedrock-runtime.us-west-2.amazonaws.com",
            "content_type": "application/json",
            "auth": "Sig_V4",
            "max_tokens_to_sample": "8000",
            "service_name": "bedrock",
            "temperature": "0.0001",
            "response_filter": "$.completion",
            "region": "us-west-2",
            "anthropic_version": "bedrock-2023-05-31"
          }
        }
      }
    ]
  }
}
```

Si el dominio reside en una nube privada virtual (Amazon VPC), debe haberse conectado a la Amazon VPC para que la solicitud cree correctamente el conector de IA. El acceso a una Amazon VPC depende de la configuración de red, pero generalmente implica conectarse a una VPN o una red corporativa. Para comprobar que tiene acceso al dominio de OpenSearch Service, diríjase a `https://`*`your-vpc-domain`*`.`*`region`*`.es.amazonaws.com` en un navegador web y compruebe que recibe la respuesta JSON predeterminada. Reemplace el *`texto de marcador`* con sus propios valores.

Cliente Python de muestra

El cliente de Python es más simple de automatizar que una solicitud HTTP y tiene una mejor reutilización. Para crear el conector de IA con el cliente Python, guarde el siguiente código de ejemplo en un archivo Python. El cliente requiere los paquetes AWS SDK para Python (Boto3), Requests:HTTP for Humans y requests-aws4auth 1.2.3.

```
import boto3
import requests
from requests_aws4auth import AWS4Auth

host = 'domain-endpoint/'
region = 'region'
service = 'es'
credentials = boto3.Session().get_credentials()
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key, region, service,
 session_token=credentials.token)

path = '_plugins/_flow_framework/workflow'
url = host + path

payload = {
  "name": "Deploy Claude Model",
  "description": "Deploy a model using a connector to Claude",
  "use_case": "PROVISION",
  "version": {
    "template": "1.0.0",
    "compatibility": [
      "2.12.0",
      "3.0.0"
    ]
  },
  "workflows": {
    "provision": {
      "nodes": [
        {
          "id": "create_claude_connector",
          "type": "create_connector",
          "user_inputs": {
            "name": "Claude Instant Runtime Connector",
            "version": "1",
            "protocol": "aws_sigv4",
            "description": "The connector to Bedrock service for Claude model",
```

```
            "actions": [
              {
                "headers": {
                  "x-amz-content-sha256": "required",
                  "content-type": "application/json"
                },
                "method": "POST",
                "request_body": "{ \"prompt\":\"${parameters.prompt}\",
 \"max_tokens_to_sample\":${parameters.max_tokens_to_sample},
 \"temperature\":${parameters.temperature},  \"anthropic_version\":
\"${parameters.anthropic_version}\" }",
                "action_type": "predict",
                "url": "https://bedrock-runtime.us-west-2.amazonaws.com/model/
anthropic.claude-instant-v1/invoke"
              }
            ],
            "credential": {
                "roleArn": "arn:aws:iam::account-id:role/opensearch-secretmanager-
role"
             },
            "parameters": {
              "endpoint": "bedrock-runtime.us-west-2.amazonaws.com",
              "content_type": "application/json",
              "auth": "Sig_V4",
              "max_tokens_to_sample": "8000",
              "service_name": "bedrock",
              "temperature": "0.0001",
              "response_filter": "$.completion",
              "region": "us-west-2",
              "anthropic_version": "bedrock-2023-05-31"
            }
          }
        }
      ]
    }
  }
}

headers = {"Content-Type": "application/json"}

r = requests.post(url, auth=awsauth, json=payload, headers=headers)
print(r.status_code)
print(r.text)
```

Plantillas de flujo de trabajo predefinidas

Amazon OpenSearch Service proporciona varias plantillas de flujo de trabajo para algunos casos de uso de machine learning (ML) comunes. El uso de una plantilla simplifica las configuraciones complejas y proporciona muchos valores predeterminados para casos de uso como la búsqueda semántica o conversacional. Puede especificar una plantilla de flujo de trabajo al llamar a la API Create Workflow.

- Para usar una plantilla de flujo de trabajo proporcionada por OpenSearch Service, especifique el caso de uso de la plantilla como parámetro de consulta `use_case`.
- Para usar una plantilla de flujo de trabajo personalizada, proporcione la plantilla completa en el cuerpo de la solicitud. Para ver un ejemplo de plantilla personalizada, consulte un ejemplo de plantilla JSON o un ejemplo de plantilla YAML.

Casos de uso de plantillas

Esta tabla proporciona información general de las diferentes plantillas disponibles, una descripción de las plantillas y los parámetros necesarios.

Caso de uso de plantillas	Descripción	Parámetros requeridos
`bedrock_titan_embedding_model_deploy`	Crea e implementa un modelo de incrustación de Amazon Bedrock (de forma predeterminada, `titan-embed-text-v1`)	`create_connector.credential.roleArn`
`bedrock_titan_embedding_model_deploy`	Crea e implementa un modelo de incrustación multimodal de Amazon Bedrock (de forma predeterminada, `titan-embed-text-v1`)	`create_connector.credential.roleArn`
`cohere_embedding_model_deploy`	Crea e implementa un modelo de incrustación de Cohere (de forma predeterminada, embed-english-v3.0).	`create_connector.credential.roleArn` , `create_connector.credential.secretArn`

Caso de uso de plantillas	Descripción	Parámetros requeridos
`cohere_chat_model_deploy`	Crea e implementa un modelo de chat de Cohere (de forma predeterminada, Cohere Command).	`create_connector.credential.roleArn`, `create_connector.credential.secretArn`
`open_ai_embedding_model_deploy`	Crea e implementa un modelo de incrustación de OpenAI (de forma predeterminada, text-embedding-ada-002).	`create_connector.credential.roleArn`, `create_connector.credential.secretArn`
`openai_chat_model_deploy`	Crea e implementa un modelo de chat de OpenAI (por defecto, gpt-3.5-turbo).	`create_connector.credential.roleArn`, `create_connector.credential.secretArn`
`semantic_search_with_cohere_embedding`	Configura la búsqueda semántica e implementa un modelo de incrustación de Cohere. Debe proporcionar la clave de API para el modelo de Cohere.	`create_connector.credential.roleArn`, `create_connector.credential.secretArn`
`semantic_search_with_cohere_embedding_query_enricher`	Configura la búsqueda semántica e implementa un modelo de incrustación de Cohere. Agrega un procesador de búsqueda query_enricher que establece un ID de modelo predeterminado para las consultas neuronales. Debe proporcionar la clave de API para el modelo de Cohere.	`create_connector.credential.roleArn`, `create_connector.credential.secretArn`

Caso de uso de plantillas	Descripción	Parámetros requeridos
`multimodal_search_with_bedrock_titan`	Implementa un modelo multimodal de Amazon Bedrock y configura una canalización de ingesta con un procesador text_image_embedding y un índice k-NN para la búsqueda multimodal. Debe proporcionar sus credenciales de AWS.	`create_connector.credential.roleArn`

> **Note**
>
> Para todas las plantillas que requieren un ARN de secreto, la opción predeterminada es almacenar el secreto con el nombre de clave “key” en AWS Secrets Manager.

Plantillas predeterminadas con modelos entrenados previamente

Amazon OpenSearch Service ofrece dos plantillas de flujo de trabajo predeterminadas adicionales que no están disponibles en OpenSearch Service de código abierto.

Caso de uso de plantillas	Descripción
`semantic_search_with_local_model`	Configura la búsqueda semántica e implementa un modelo entrenado previamente (`msmarco-distilbert-base-tas-b`). Agrega un procesador de búsqueda `neural_query_enricher` que establece un ID de modelo predeterminado para las consultas neuronales y crea un índice k-NN vinculado denominado “my-nlp-index”.
`hybrid_search_with_local_model`	Configura la búsqueda híbrida e implementa un modelo entrenado previamente (`msmarco-distilbert-base-tas-b`). Agrega un procesador de búsqueda `neural_qu`

Caso de uso de plantillas	Descripción
	ery_enricher que establece un ID de modelo predeterminado para las consultas neuronales y crea un índice k-NN vinculado denominado “my-nlp-index”.

Configuración de permisos

Si crea un dominio nuevo con la versión 2.13 o una posterior, los permisos ya están en vigor. Si habilita Flow Framework en un dominio preexistente de OpenSearch Service con la versión 2.11 o anterior y, a continuación, lo actualiza a la versión 2.13 o una posterior, debe definir el rol `flow_framework_manager`. Los usuarios que no sean administradores deben estar asignados a este rol para poder administrar índices templados en los dominios mediante un control de acceso detallado. Para crear el rol `flow_framework_manager` de forma manual, siga estos pasos:

1. En OpenSearch Dashboards, vaya a Seguridad y elija Permisos.
2. Seleccione Crear grupo de acciones y configure los siguientes grupos:

Nombre del grupo	Permisos
`flow_framework_full_access`	• `cluster:admin/opensearch/flow_framework/*` • `cluster_monitor`
`flow_framework_read_accesss`	• `cluster:admin/opensearch/flow_framework/workflow/get` • `cluster:admin/opensearch/flow_framework/workflow/search` • `cluster:admin/opensearch/flow_framework/workflow_state/get` • `cluster:admin/opensearch/flow_framework/workflow_state/search`

3. Seleccione Roles y, a continuación, Crear rol.
4. Asigne el nombre flow_framework_manager al rol.

5. Para Permisos de clúster, seleccione `flow_framework_full_access` y `flow_framework_read_access`.
6. Para Índice, escriba `*`.
7. Para Permisos de índice, seleccione `indices:admin/aliases/get`, `indices:admin/mappings/get`, y `indices_monitor`.
8. Elija Crear.
9. Después de crear el rol, debe asignarlo a cualquier rol de usuario o backend que administre los índices de Flow Framework.

Security Analytics para Amazon OpenSearch Service

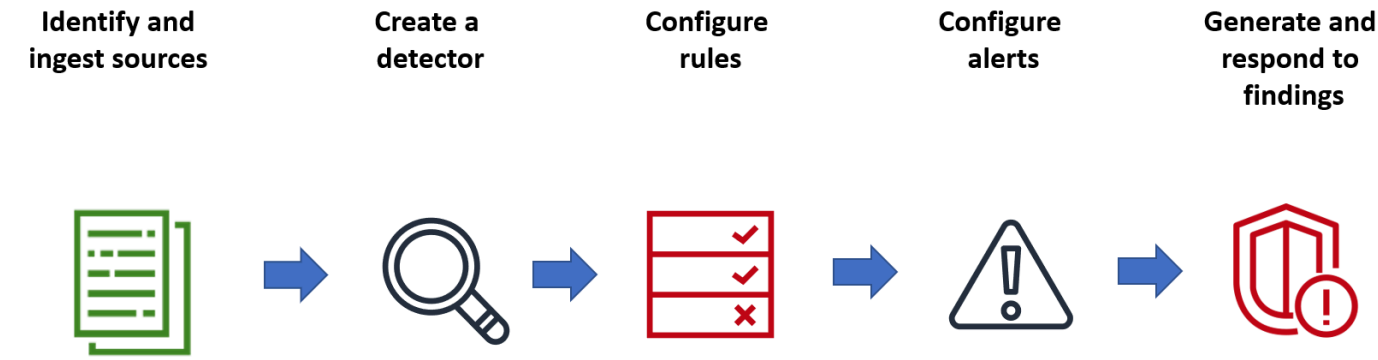
Security Analytics es una solución de OpenSearch que proporciona visibilidad de la infraestructura de su organización, monitorea las actividades anómalas, detecta posibles amenazas a la seguridad en tiempo real y activa alertas a destinos preconfigurados. Puede monitorear la actividad maliciosa de sus registros de eventos de seguridad evaluando continuamente las reglas de seguridad y revisando los resultados de seguridad generados automáticamente. Además, Security Analytics puede generar alertas automatizadas y enviarlas a un canal de notificación específico, como Slack o el correo electrónico.

Puede usar el complemento de Security Analytics para detectar amenazas comunes de forma inmediata y generar información de seguridad fundamental a partir de los registros de eventos de seguridad existentes, como los registros de firewall, los registros de Windows y los registros de auditoría de autenticación. Para Security Analytics, el dominio debe ejecutarse en una versión de OpenSearch 2.5 o posterior.

> **Note**
>
> Esta documentación proporciona una breve información general de Security Analytics en Amazon OpenSearch Service. Define los conceptos clave y proporciona los pasos para configurar los permisos. Para obtener la documentación completa, que incluye una guía de configuración, una referencia de la API y una referencia de toda la configuración disponible, consulte Security Analytics en la documentación de OpenSearch.

Componentes y conceptos de Security Analytics

Una serie de herramientas y características constituyen la base del funcionamiento de Security Analytics. Los principales componentes que componen el complemento incluyen los detectores, los tipos de registro, las reglas, los resultados y las alertas.



Tipos de registro

OpenSearch admite varios tipos de registros y proporciona mapeos listos para usar para cada tipo. Al crear un detector, debe especificar el tipo de registro y configurar un intervalo de tiempo y, a partir de ahí, Security Analytics activa automáticamente un conjunto de reglas relevante que se ejecutan en ese intervalo.

Detectores

Los detectores identifican una variedad de amenazas de ciberseguridad para un tipo de registro en todos sus índices de datos. El detector se configura para que utilice tanto reglas personalizadas como reglas Sigma preconfiguradas que evalúan los eventos que ocurren en el sistema. A continuación, el detector genera los resultados de seguridad a partir de estos eventos. Para obtener más información sobre los detectores, consulte Creación de detectores en la documentación de OpenSearch.

Reglas

Las reglas de detección de amenazas definen las condiciones que los detectores aplican a los datos de registro incorporados para identificar un evento de seguridad. Security Analytics permite importar, crear y personalizar reglas para cumplir con sus requisitos, y también proporciona reglas Sigma empaquetadas previamente y de código abierto para detectar las amenazas más comunes en sus registros. Security Analytics asigna muchas reglas a una base de conocimientos cada vez mayor sobre tácticas y técnicas de los adversarios, mantenida por la organización MITRE ATT&CK. Puede usar OpenSearch Dashboards o las API para crear y usar las reglas. Para obtener más información sobre las reglas, consulte Working with rules en la documentación de OpenSearch.

Resultados

Cuando un detector hace coincidir una regla con un evento de registro, genera un resultado. Cada resultado incluye una combinación única de reglas seleccionadas, un tipo de registro y la gravedad de la regla. Los resultados no necesariamente apuntan a amenazas inminentes dentro del sistema, pero siempre aíslan un evento de interés. Para obtener más información sobre los resultados, consulte Trabajar con resultados en la documentación de OpenSearch.

Alertas

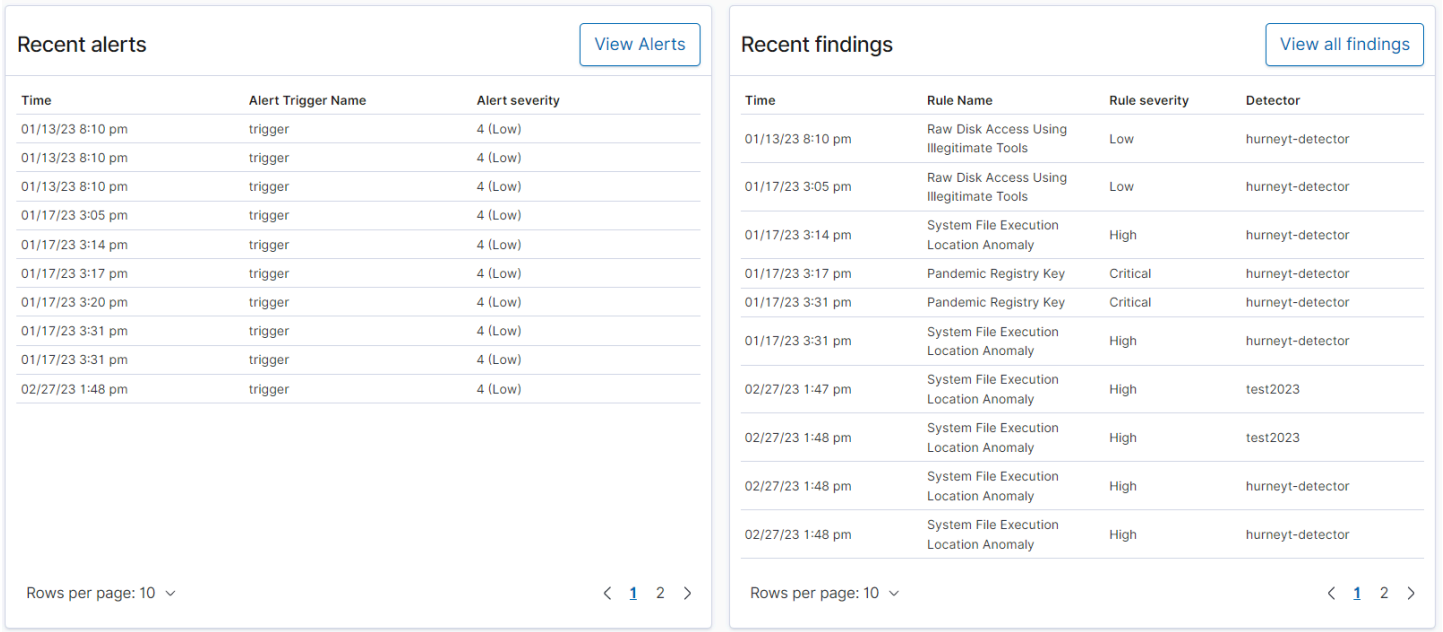
Al crear un detector, puede especificar una o varias condiciones que activan una alerta. Una alerta es una notificación que se envía a un canal preferido, como Slack o el correo electrónico. Puede configurar la alerta para que se active cuando el detector coincida con una o varias reglas y puede personalizar el mensaje de notificación. Para obtener más información sobre las alertas, consulte Trabajar con alertas en la documentación de OpenSearch.

Exploración de Security Analytics

Puede usar OpenSearch Dashboards para visualizar y obtener información sobre el complemento de Security Analytics. La vista Información general proporciona información como resultados y recuentos de alertas, resultados y alertas recientes, reglas de detección frecuentes y una lista de detectores. Puede ver una vista resumida compuesta por varias visualizaciones. El siguiente gráfico, por ejemplo, muestra la tendencia de los resultados y las alertas de varios tipos de registros durante un período de tiempo determinado.



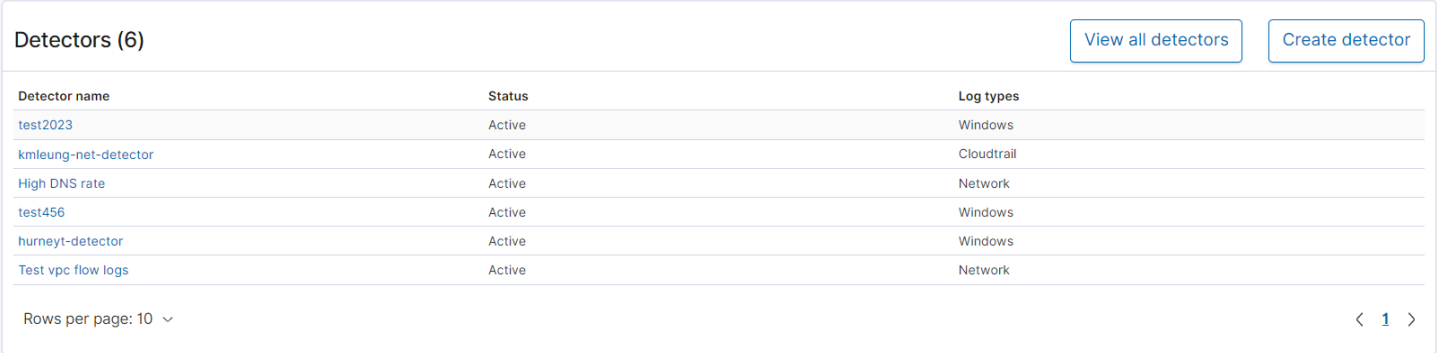
Más abajo en la página, puede revisar los resultados y alertas más recientes.



Además, puede ver una distribución de las reglas que se activan con más frecuencia en todos los detectores activos. Esto puede ayudarle a detectar e investigar diferentes tipos de actividades maliciosas en todos los tipos de registros.



Por último, puede ver el estado de los detectores configurados. Desde este panel, también puede acceder al flujo de trabajo de creación de detectores.



Para configurar Security Analytics, cree reglas en la página de Reglas y utilícelas para escribir los detectores en la página de Detectores. Para tener una visión más precisa de los resultados de Security Analytics, puede utilizar las páginas de Resultados y Alertas.

Configuración de permisos

Si habilita Security Analytics en un dominio de OpenSearch Service preexistente, el rol `security_analytics_manager` podría no estar definido en el dominio. Los usuarios que no sean administradores deben estar asignados a este rol para poder administrar índices templados en los dominios mediante un control de acceso detallado. Para crear el rol `security_analytics_manager` de forma manual, siga estos pasos:

1. En OpenSearch Dashboards, vaya a Seguridad y elija Permisos.
2. Seleccione Crear grupo de acciones y configure los siguientes grupos:

Nombre del grupo	Permisos
security_analytics_full_access	• cluster:admin/opensearch/securityanalytics/alerts/* • cluster:admin/opensearch/securityanalytics/detector/* • cluster:admin/opensearch/securityanalytics/findings/* • cluster:admin/opensearch/securityanalytics/mapping/* • cluster:admin/opensearch/securityanalytics/rule/*
security_analytics_read_access	• cluster:admin/opensearch/securityanalytics/alerts/get • cluster:admin/opensearch/securityanalytics/detector/get • cluster:admin/opensearch/securityanalytics/detector/search • cluster:admin/opensearch/securityanalytics/findings/get • cluster:admin/opensearch/securityanalytics/mapping/get • cluster:admin/opensearch/securityanalytics/mapping/view/get • cluster:admin/opensearch/securityanalytics/rule/get • cluster:admin/opensearch/securityanalytics/rule/search

3. Seleccione Roles y, a continuación, Crear rol.
4. Asigne el nombre security_analytics_manager al rol.

5. Para Permisos de clúster, seleccione `security_analytics_full_access` y `security_analytics_read_access`.
6. Para Índice, escriba `*`.
7. Para Permisos de índice, seleccione `indices:admin/mapping/put` y `indices:admin/mappings/get`.
8. Seleccione Crear.
9. Después de crear el rol, debe mapearlo a cualquier rol de usuario o backend que administre los índices de Security Analytics.

Solución de problemas

No existe tal error de índice

Si no tiene detectores y abre el panel de Security Analytics, es posible que vea una notificación en la parte inferior derecha que diga `[index_not_found_exception] no such index [.opensearch-sap-detectors-config]`. Puede ignorar esta notificación, que desaparece en unos segundos y no volverá a aparecer una vez que crea un detector.

Observabilidad en Amazon OpenSearch Service

La instalación predeterminada de OpenSearch Dashboards para Amazon OpenSearch Service incluye el complemento Observability, que se puede utilizar para visualizar eventos controlados por datos mediante el lenguaje de procesamiento con plecas (PPL) para explorar, descubrir y consultar los datos almacenados en OpenSearch. Este complemento requiere OpenSearch 1.2 o posterior.

El complemento Observability proporciona una experiencia unificada para recopilar y supervisar métricas, registros y seguimientos de orígenes de datos habituales. La recopilación y supervisión de datos en un solo lugar permite una observabilidad integral de extremo a extremo de toda la infraestructura.

> **Note**
>
> Esta documentación proporciona una breve descripción de Observability en OpenSearch Service. Para obtener la documentación completa del complemento Observability, incluidos los permisos, consulte Observability.

Cada cual tiene un proceso de exploración de datos diferente. Si no tiene experiencia con la exploración de datos y la creación de visualizaciones, le recomendamos que pruebe un flujo de trabajo como el siguiente:

Explore los datos con análisis de eventos

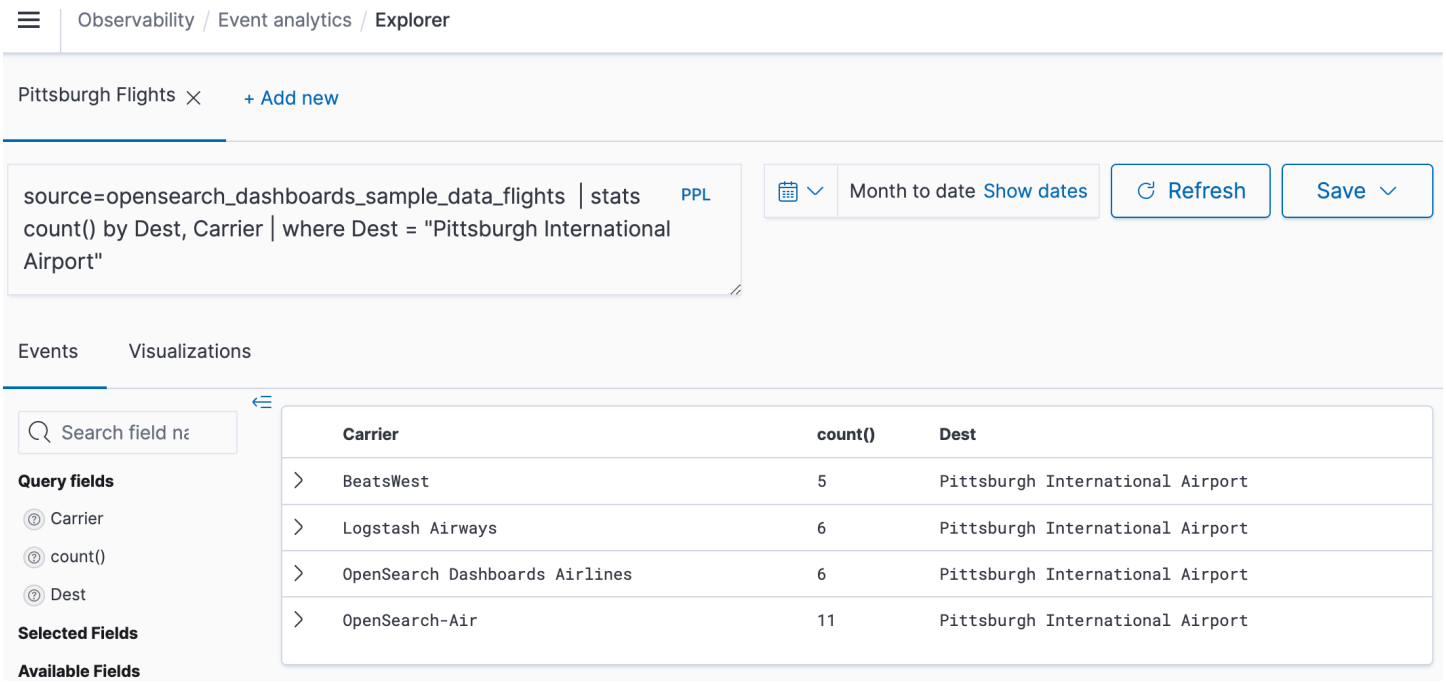
Para empezar, supongamos que está recopilando datos de vuelos en un dominio de OpenSearch Service y desea averiguar qué aerolínea tuvo más llegadas de vuelos desde el Aeropuerto Internacional de Pittsburgh el mes pasado. Escribe la siguiente consulta PPL:

```
source=opensearch_dashboards_sample_data_flights |
    stats count() by Dest, Carrier |
    where Dest = "Pittsburgh International Airport"
```

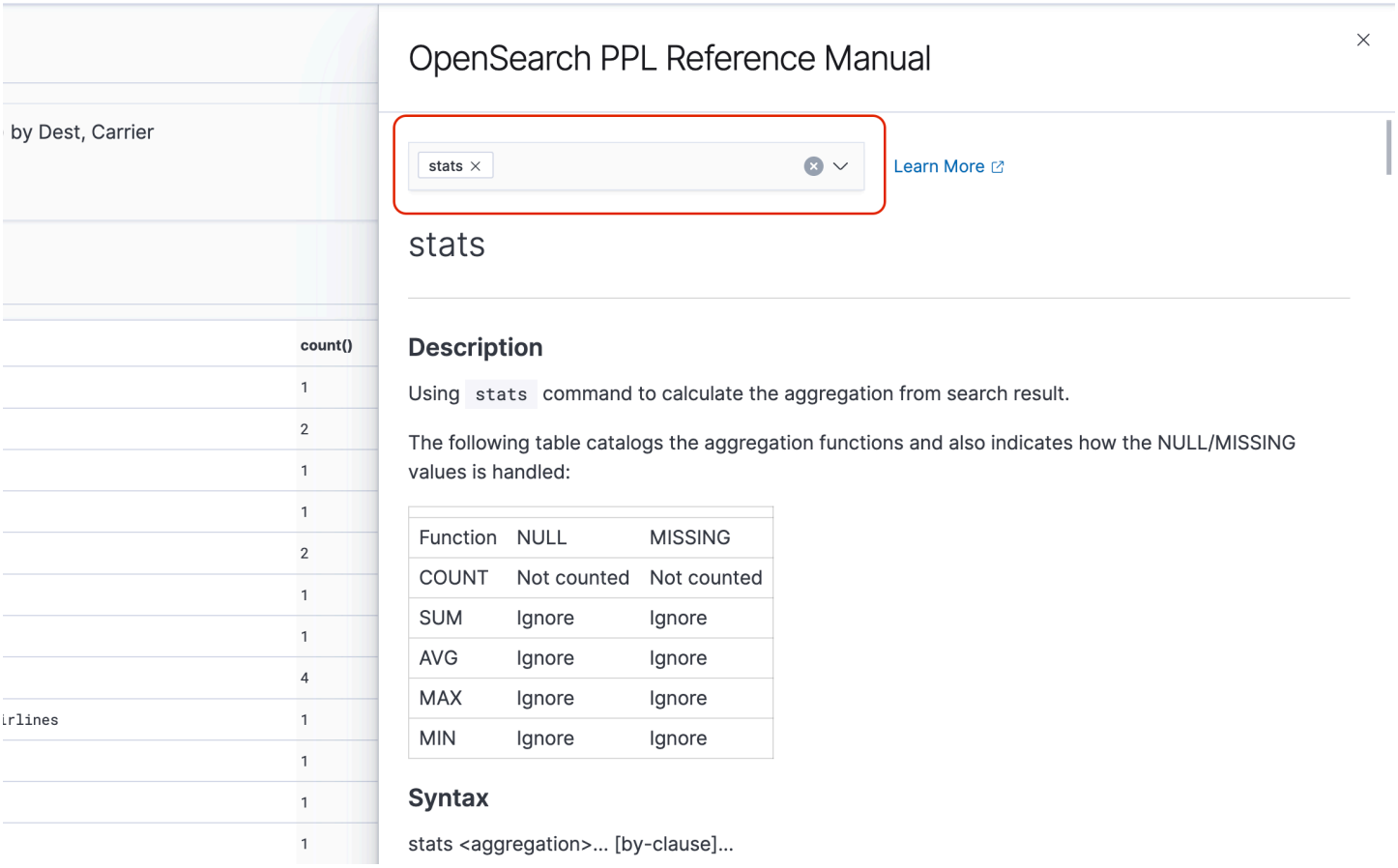
Esta consulta extrae datos del índice denominado `opensearch_dashboards_sample_data_flights`. A continuación, utiliza el comando `stats` para obtener un recuento total de vuelos y agrupar por aeropuerto de destino y aerolínea. Por último,

utiliza la cláusula `where` para filtrar los resultados y mostrar los vuelos con llegada al Aeropuerto Internacional de Pittsburgh.

Este es el aspecto de los datos que se muestran para el último mes:



Puede elegir el botón PPL del editor de consultas para obtener información y ejemplos de uso de cada comando PPL:



Veamos un ejemplo más complejo, en el que se realizan consultas de información sobre retrasos de vuelos:

```
source=opensearch_dashboards_sample_data_flights |
    where FlightDelayMin > 0 |
    stats sum(FlightDelayMin) as minimum_delay, count() as total_delayed by Carrier,
 Dest |
    eval avg_delay=minimum_delay / total_delayed |
    sort - avg_delay
```

Cada comando de la consulta afecta al resultado final:

- `source=opensearch_dashboards_sample_data_flights`: extrae datos del mismo índice que el ejemplo anterior
- `where FlightDelayMin > 0`: filtra los datos para mostrar los vuelos retrasados

- `stats sum(FlightDelayMin) as minimum_delay, count() as total_delayed by Carrier`: para cada aerolínea, obtiene el tiempo de retraso mínimo total y el recuento total de vuelos retrasados
- `eval avg_delay=minimum_delay / total_delayed`: calcula el tiempo de retraso medio de cada aerolínea dividiendo el tiempo de retraso mínimo entre el número total de vuelos retrasados
- `sort - avg_delay`: ordena los resultados por retraso medio en orden descendente

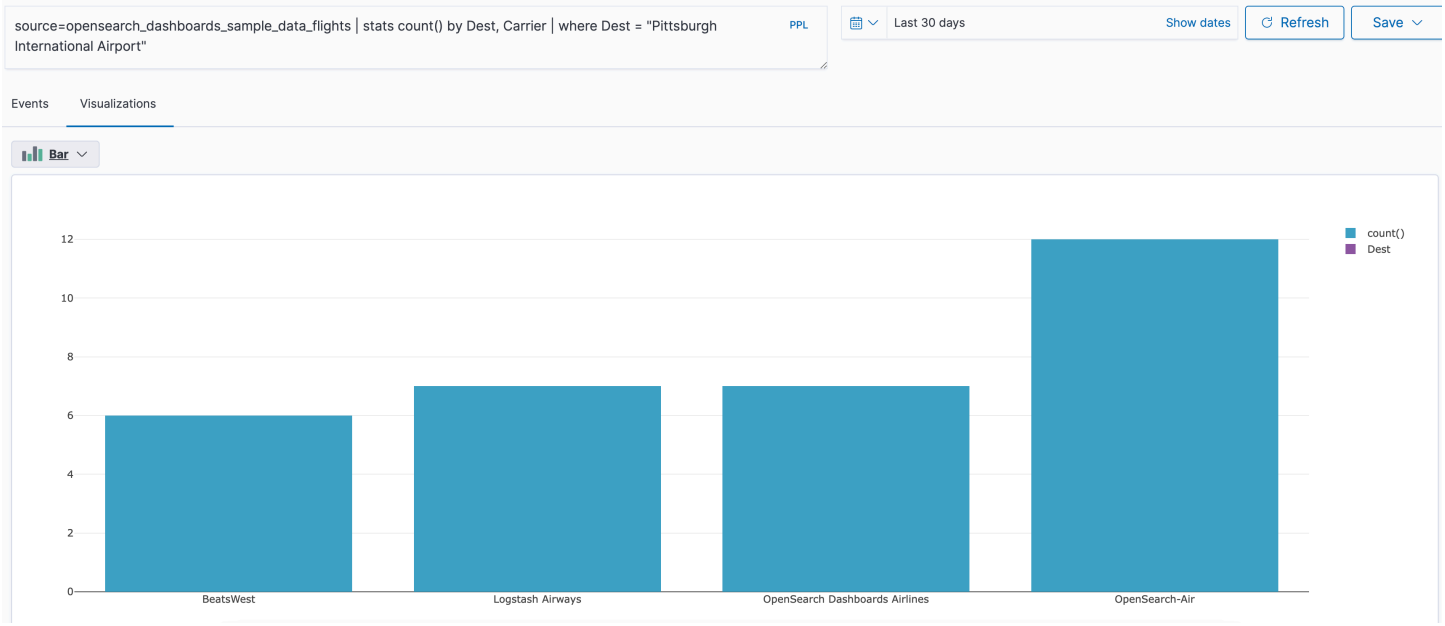
Con esta consulta, se puede determinar que OpenSearch Dashboards Airlines tiene menos retrasos de media.

	avg_delay	Carrier	minimum_delay	total_delayed
>	212	Logstash Airways	4470	21
>	184	OpenSearch-Air	4245	23
>	155	BeatsWest	2025	13
>	153	OpenSearch Dashboards Airlines	4305	28

Dispone de más ejemplos de consultas PPL en Queries and Visualizations (Consultas y visualizaciones) en la página Event analytics (Análisis de eventos).

Crear visualizaciones

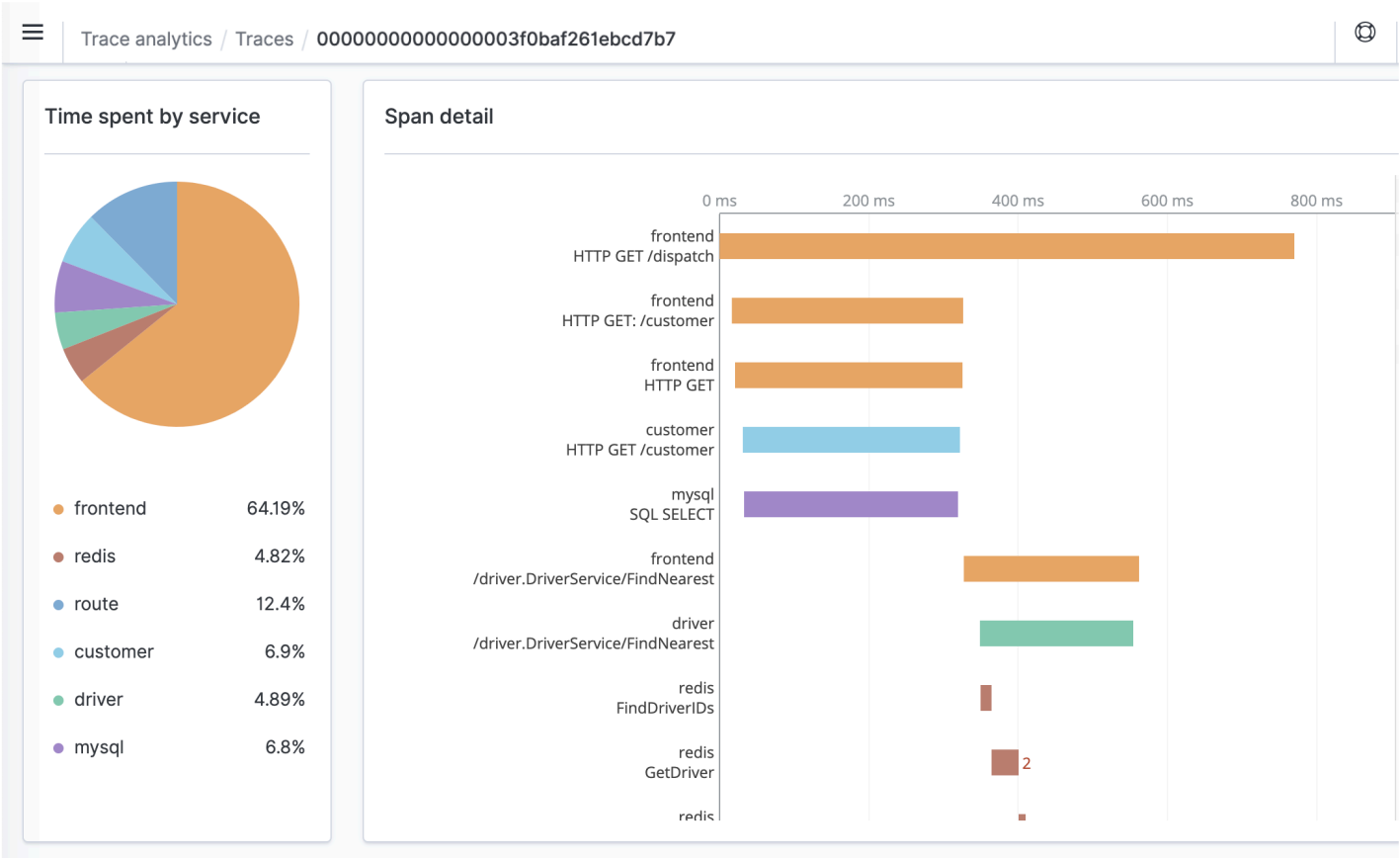
Una vez que haya consultado correctamente los datos que le interesan, puede guardar esas consultas como visualizaciones:



A continuación, agregue esas visualizaciones a paneles operacionales para comparar diferentes datos. Aproveche los cuadernos para combinar diferentes visualizaciones y bloques de código y compartirlos con miembros de su equipo.

Profundizar más con Trace Analytics

Trace Analytics proporciona una forma de visualizar el flujo de eventos de los datos de OpenSearch para identificar y solucionar problemas de rendimiento en aplicaciones distribuidas.



Análisis de seguimiento para Amazon OpenSearch Service

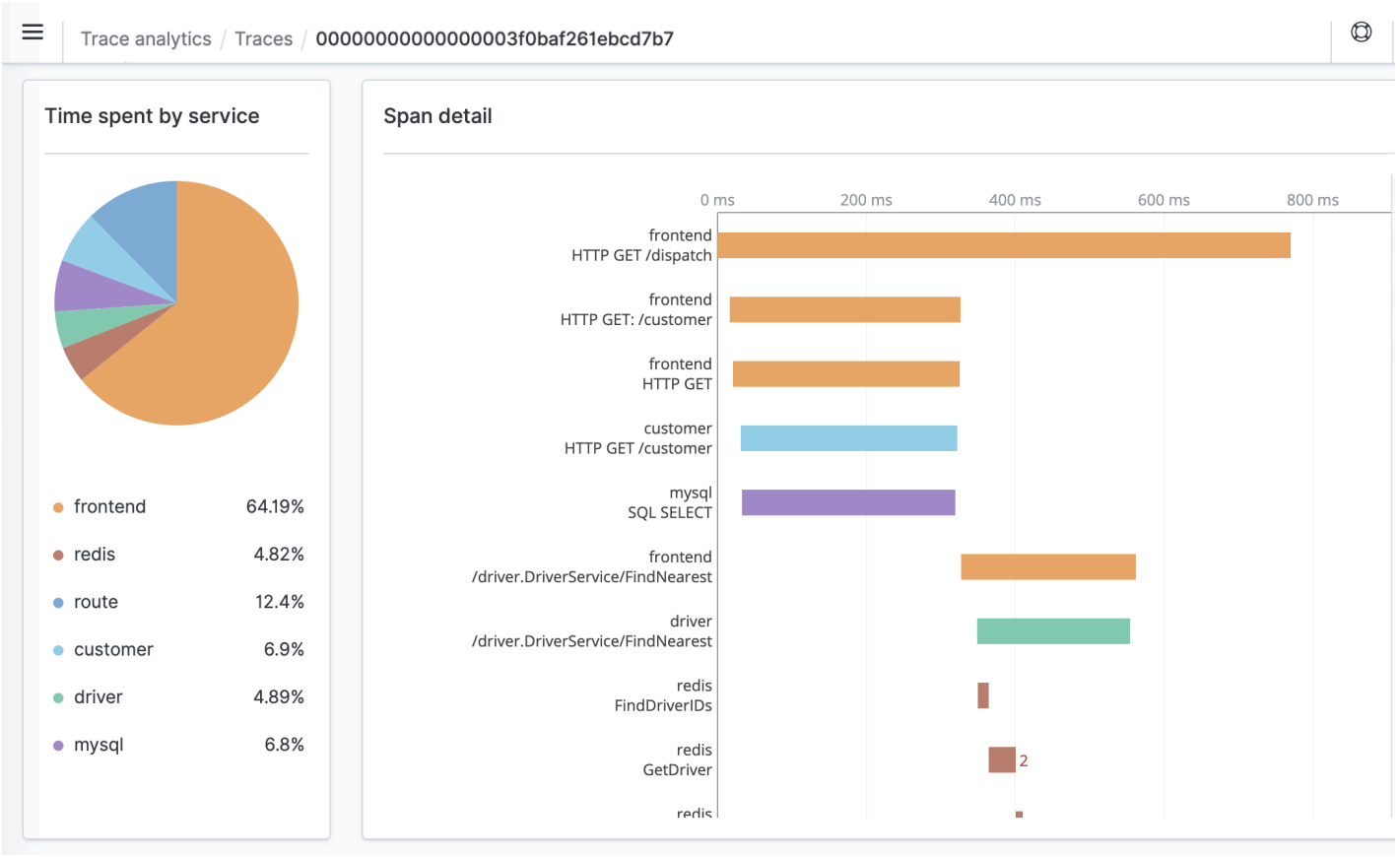
Puede utilizar Trace Analytics, que forma parte del complemento Observability de OpenSearch, para analizar datos de seguimiento de aplicaciones distribuidas. Trace Analytics requiere OpenSearch o Elasticsearch 7.9 o posterior.

En una aplicación distribuida, una sola operación, como un usuario que hace clic en un botón, puede desencadenar una serie extendida de eventos. Por ejemplo, el front-end de la aplicación puede llamar a un servicio backend, que llama a otro servicio, que consulta una base de datos, que procesa

la consulta y devuelve un resultado. Luego, el primer servicio backend envía una confirmación al front-end, que actualiza la interfaz de usuario.

Puede utilizar Trace Analytics para ayudarle a visualizar este flujo de eventos e identificar problemas de rendimiento.

> **Note**
>
> Esta documentación proporciona una breve información general de Trace Analytics. Para obtener la documentación completa, consulte Trace Analytics en la documentación de OpenSearch de código abierto.



Requisitos previos

Trace Analytics requiere que agregue instrumentación a su aplicación y genere datos de rastreo usando una biblioteca compatible con OpenTelemetry como Jaeger or Zipkin. Este paso se produce completamente fuera de OpenSearch Service. La AWSDocumentación de la Distro para

OpenTelemetry contiene aplicaciones de ejemplo para muchos lenguajes de programación que pueden ayudarle a comenzar, que incluyen Java, Python, Go y JavaScript.

Después de agregar la instrumentación a la aplicación, el Recopilador de OpenTelemetry recibe datos de la aplicación y los formatea en datos de OpenTelemetry. Vea la lista de receptores en GitHub.AWS La distribución para OpenTelemetry incluye un Receptor para AWS X-Ray.

Por último, puede usar Amazon OpenSearch Ingestion para dar formato a los datos de OpenTelemetry para su uso con OpenSearch.

Configuración de ejemplo de OpenTelemetry Collector

Para utilizar OpenTelemetry Collector con Amazon OpenSearch Ingestion, pruebe la siguiente configuración de ejemplo:

```
extensions:
  sigv4auth:
    region: "us-east-1"
    service: "osis"

receivers:
  jaeger:
    protocols:
      grpc:

exporters:
  otlphttp:
    traces_endpoint: "https://pipeline-endpoint.us-east-1.osis.amazonaws.com/
opentelemetry.proto.collector.trace.v1.TraceService/Export"
    auth:
      authenticator: sigv4auth
    compression: none

service:
  extensions: [sigv4auth]
  pipelines:
    traces:
      receivers: [jaeger]
      exporters: [otlphttp]
```

Configuración de muestra de OpenSearch Ingestion

Para enviar datos de seguimiento a un dominio de servicio OpenSearch, pruebe la siguiente configuración de canalización de OpenSearch Ingestion de ejemplo. Para obtener instrucciones a fin de crear una canalización, consulte the section called “Creación de canalizaciones”.

```
version: "2"
otel-trace-pipeline:
  source:
    otel_trace_source:
      "/${pipelineName}/ingest"
  processor:
    - trace_peer_forwarder:
  sink:
    - pipeline:
        name: "trace_pipeline"
    - pipeline:
        name: "service_map_pipeline"
trace-pipeline:
  source:
    pipeline:
      name: "otel-trace-pipeline"
  processor:
    - otel_traces:
  sink:
    - opensearch:
        hosts: ["https://domain-endpoint"]
        index_type: trace-analytics-raw
        aws:
          # IAM role that OpenSearch Ingestion assumes to access the domain sink
          sts_role_arn: "arn:aws:iam::{account-id}:role/pipeline-role"
          region: "us-east-1"

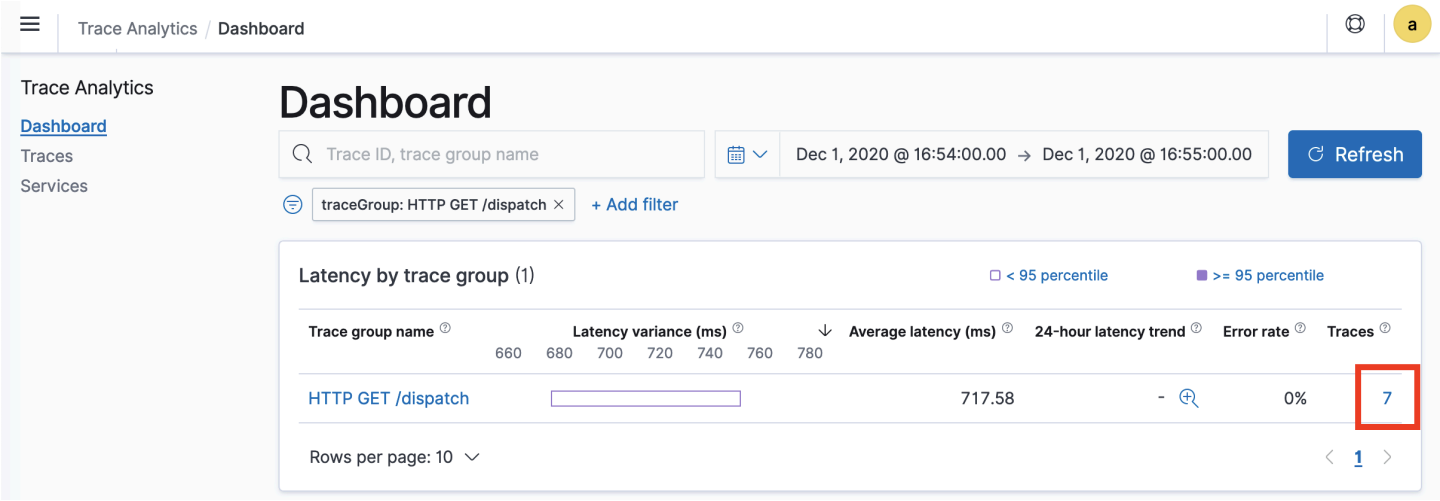
service-map-pipeline:
  source:
    pipeline:
      name: "otel-trace-pipeline"
  processor:
    - service_map:
  sink:
    - opensearch:
        hosts: ["https://domain-endpoint"]
        index_type: trace-analytics-service-map
```

```
    aws:
      # IAM role that the pipeline assumes to access the domain sink
      sts_role_arn: "arn:aws:iam::{account-id}:role/pipeline-role"
      region: "us-east-1"
```

El rol de canalización que especifique en la opción `sts_role_arn` debe disponer de permisos de escritura en el receptor. Para obtener instrucciones sobre cómo configurar los permisos para el rol de la canalización, consulte the section called “Configuración de roles y usuarios”.

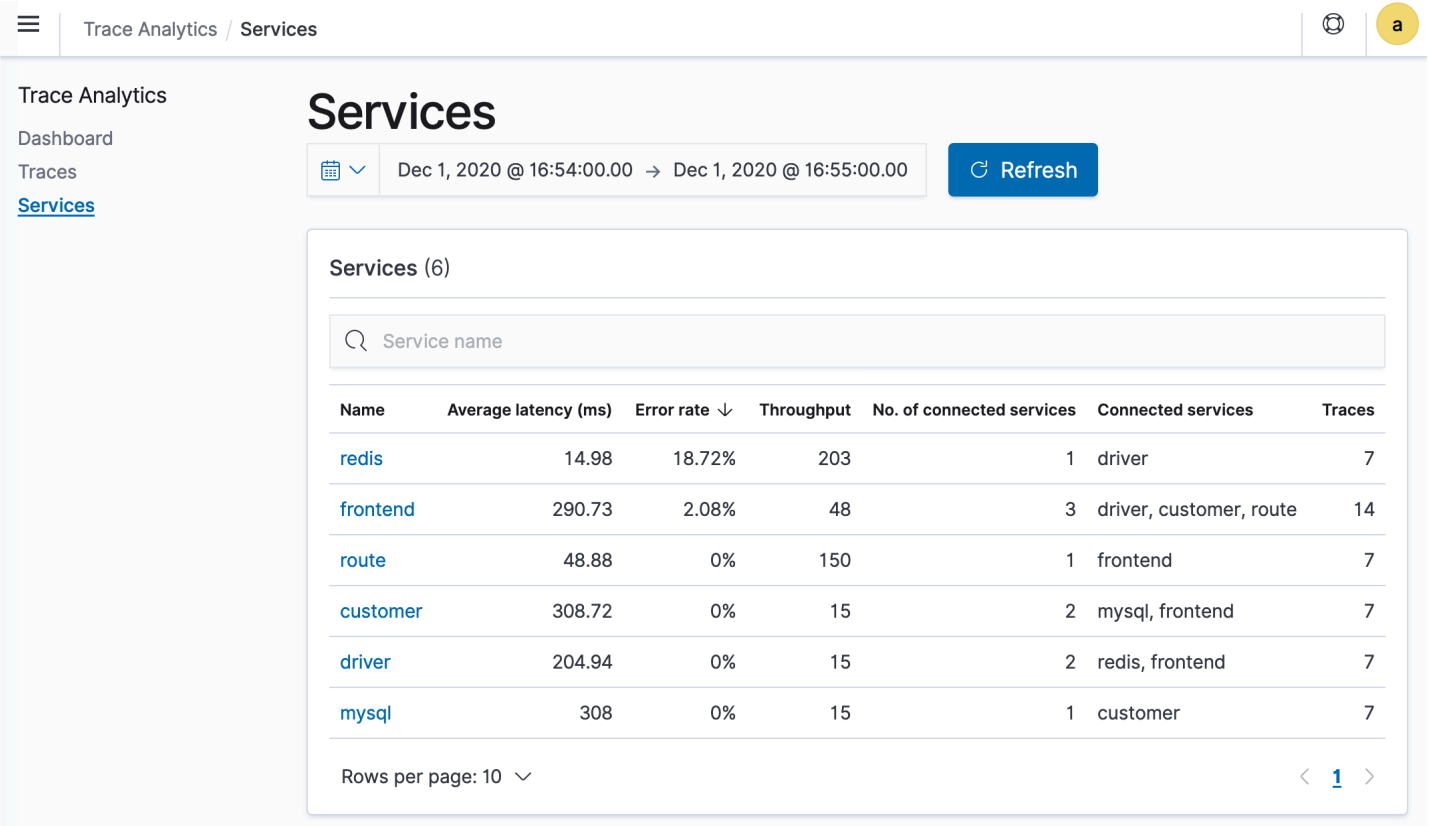
Exploración de datos de rastreo

La vista de Panel agrupa los rastreos por método HTTP y ruta de acceso para que pueda ver la latencia media, la tasa de error y las tendencias asociadas a una operación concreta. Para obtener una vista más enfocada, intente filtrar por nombre de grupo de rastreo.



Para detallar los rastreos que componen un grupo de rastreo, elija el número de rastreos en la columna de la derecha. A continuación, elija un rastreo individual para obtener un resumen detallado.

La vista de Servicios contiene todos los servicios de la aplicación, además de un mapa interactivo que muestra cómo los distintos servicios se conectan entre sí. A diferencia del panel (que ayuda a identificar problemas por operación), el mapa de servicio le ayuda a identificar problemas por servicio. Intente ordenar por tasa de error o latencia para obtener una idea de las áreas problemáticas potenciales de su aplicación.



Consulta de datos de Amazon OpenSearch Service mediante el lenguaje de procesamiento de canalizaciones

El lenguaje de procesamiento de canalizaciones (PPL) es un lenguaje de consulta que permite utilizar la sintaxis de canalizaciones (|) para consultar los datos almacenados en Amazon OpenSearch Service. La PPL requiere OpenSearch o Elasticsearch 7.9 o posterior.

> **Note**
>
> Esta documentación proporciona una breve información general de PPL en Amazon OpenSearch Service. Para ver los pasos detallados y una referencia completa de los comandos, consulte PPL en la documentación de código abierto de OpenSearch.

La sintaxis PPL consta de comandos delimitados por un carácter de canalización (|) donde los datos fluyen de izquierda a derecha a través de cada canalización. Por ejemplo, la sintaxis PPL para buscar el número de anfitriones con errores HTTP 403 o 503, agregarlos por anfitrión y ordenarlos en el orden de impacto es la siguiente:

```
source = dashboards_sample_data_logs | where response='403' or response='503' | stats
 count(request) as request_count by host, response | sort -request_count
```

Para comenzar, elija Query Workbench (Consulta del entorno de trabajo) en OpenSearch Dashboards y seleccione PPL. Utilice la operación `bulk` para indexar algunos datos de muestra:

```
PUT accounts/_bulk?refresh
{"index":{"_id":"1"}}
{"account_number":1,"balance":39225,"firstname":"Amber","lastname":"Duke","age":32,"gender":"M"
 Holmes
 Lane","employer":"Pyrami","email":"amberduke@pyrami.com","city":"Brogan","state":"IL"}
{"index":{"_id":"6"}}
{"account_number":6,"balance":5686,"firstname":"Hattie","lastname":"Bond","age":36,"gender":"M"
 Bristol
 Street","employer":"Netagy","email":"hattiebond@netagy.com","city":"Dante","state":"TN"}
{"index":{"_id":"13"}}
{"account_number":13,"balance":32838,"firstname":"Nanette","lastname":"Bates","age":28,"gender"
 Mady Street","employer":"Quility","city":"Nogal","state":"VA"}
{"index":{"_id":"18"}}
{"account_number":18,"balance":4180,"firstname":"Dale","lastname":"Adams","age":33,"gender":"M"
 Hutchinson Court","email":"daleadams@boink.com","city":"Orick","state":"MD"}
```

En el siguiente ejemplo, se devuelven los campos `firstname` y `lastname` para documentos en un índice de cuentas con `age`: mayor que 18:

```
search source=accounts | where age > 18 | fields firstname, lastname
```

Respuesta de ejemplo

id	firstname	lastname
0	Amber	Duke
1	Hattie	Bond
2	Nanette	Bates
3	Dale	Adams

Puede utilizar un conjunto completo de comandos de solo lectura como `search`, `where`, `fields`, `rename`, `dedup`, `stats`, `sort`, `eval`, `head`, `top` `yrare`. El complemento PPL admite todas las funciones SQL, incluidas las matemáticas, las trigonométricas, las de fecha y hora, las de cadena, las agregadas, las de operadores y expresiones avanzadas. Para obtener más información, consulte el OpenSearch PPL reference manual (Manual de referencia de PPL de OpenSearch).

Prácticas recomendadas operativas para Amazon OpenSearch Service

En este capítulo se proporcionan prácticas recomendadas sobre la operación de dominios de Amazon OpenSearch Service e incluye instrucciones generales que se aplican a muchos casos de uso. Cada carga de trabajo es única, con características únicas, por lo que ninguna recomendación genérica es exactamente adecuada para cada caso de uso. La práctica general más importante consiste en implementar, probar y ajustar sus dominios en un ciclo continuo para encontrar la configuración, la estabilidad y el costo óptimos para su carga de trabajo.

Temas

- Monitorización y alertas
- Estrategia de particiones
- Stability
- Rendimiento
- Seguridad
- Optimización de costos
- Tamaño de los dominios de Amazon OpenSearch Service
- Escala de petabytes en Amazon OpenSearch Service
- Nodos coordinadores dedicados en Amazon OpenSearch Service
- Nodos maestros dedicados en Amazon OpenSearch Service

Monitorización y alertas

Las siguientes prácticas recomendadas se aplican al monitoreo de los dominios de OpenSearch Service.

Configuración de alarmas de CloudWatch

OpenSearch Service emite métricas de rendimiento a Amazon CloudWatch. Revise regularmente sus métricas de clústeres e instancias y configure las alarmas de CloudWatch recomendadas según el rendimiento de la carga de trabajo.

Habilitación de la publicación de registros

OpenSearch Service expone registros de errores, registros lentos de búsqueda, registros lentos de índice y registros de auditoría de OpenSearch Service en Registros de Amazon CloudWatch. Los registros lentos de búsqueda, los registros lentos de índice y los registros de errores son útiles para la resolución de problemas de rendimiento y estabilidad. Los registros de auditoría, que solo están disponibles si habilita control de acceso preciso, realizan un seguimiento de la actividad del usuario. Para más información, consulte Registros en la documentación de OpenSearch.

Los registros lentos de búsqueda e indexación de registros lentos son una herramienta importante para comprender y solucionar problemas del rendimiento de sus operaciones de búsqueda e indexación. Habilite la entrega lenta de registros de búsqueda e índice para todos los dominios de producción. También debe configurar umbrales de registro, de lo contrario, CloudWatch no capturará los registros.

Estrategia de particiones

Las particiones distribuyen la carga de trabajo en los nodos de datos de su dominio de OpenSearch Service. Los índices configurados correctamente pueden ayudar a mejorar el rendimiento general del dominio.

Cuando envía datos a OpenSearch Service, envía esos datos a un índice. Un índice es análogo a una tabla de base de datos, con documentos como las filas y campos como las columnas. Cuando crea el índice, le dice a OpenSearch cuántas particiones principales desea crear. Los fragmentos principales son particiones independientes del conjunto de datos completo. OpenSearch Service distribuye automáticamente los datos entre las particiones principales de un índice. También puede configurar réplicas del índice. Cada partición de réplica incluye un conjunto completo de copias de las particiones principales de ese índice.

OpenSearch Service asigna las particiones de cada índice en los nodos de datos del clúster. Garantiza que las particiones principales y de réplica del índice residan en nodos de datos diferentes. La primera réplica garantiza que tenga dos copias de los datos en el índice. Siempre debe usar al menos una réplica. Las réplicas adicionales proporcionan redundancia y capacidad de lectura adicionales.

OpenSearch envía solicitudes de índice a todos los nodos de datos que contienen particiones que pertenecen al índice. Envía solicitudes de indexación primero a los nodos de datos que contienen particiones principales y, después, a los nodos de datos que contienen particiones de réplica. El nodo

coordinador dirige las solicitudes de búsqueda, ya sea a una partición principal o a una réplica, para todas las particiones que pertenecen al índice.

Por ejemplo, para un índice con cinco particiones principales y una réplica, cada solicitud de indexación toca 10 particiones. En contraste, las solicitudes de búsqueda se envían a las particiones n, donde n es el número de particiones principales. Para un índice con cinco particiones principales y una réplica, cada consulta de búsqueda toca cinco particiones (principales o réplicas) de ese índice.

Determinar los recuentos de particiones y nodos de datos

Utilice las siguientes prácticas recomendadas para determinar los recuentos de nodos de datos y particiones para su dominio.

Tamaño de las particiones: el tamaño de los datos en el disco es un resultado directo del tamaño de los datos de origen y cambia a medida que se indexan más datos. La relación de fuente a índice puede variar enormemente, de 1:10 a 10:1 o más, pero por lo general es de alrededor de 1:1.10. Puede usar esa relación para predecir el tamaño del índice en el disco. Puede también indexar algunos datos y recuperar los tamaños reales del índice para determinar la relación de la carga de trabajo. Después de tener un tamaño de índice previsto, establezca un recuento de particiones para que cada partición tenga entre 10 y 30 GiB (para cargas de trabajo de búsqueda) o entre 30 y 50 GiB (para cargas de trabajo de registros). 50 GiB debería ser el máximo; asegúrese de planificar el crecimiento.

Recuento de particiones: la distribución de particiones a los nodos de datos tiene un gran impacto en el rendimiento de un dominio. Cuando tenga índices con múltiples particiones, procure que la cantidad de particiones sea un múltiplo de la cantidad de nodos de datos. Esto ayuda a garantizar que las particiones se distribuyan de manera uniforme entre los nodos de datos y evita los nodos activos. Por ejemplo, si tiene 12 particiones principales, el recuento de nodos de datos debe ser 2, 3, 4, 6 o 12. Sin embargo, el recuento de particiones es secundario al tamaño de la partición; si tiene 5 GiB de datos, debe seguir utilizando una sola partición.

Particiones por nodo de datos: el número total de particiones que puede contener un nodo es proporcional a la memoria en montón de máquina virtual Java (JVM) del nodo. Trate de obtener 25 particiones o menos por GiB de memoria en montón. Por ejemplo, un nodo con 32 GiB de memoria en montón no debe contener más de 800 particiones. Aunque la distribución de particiones puede variar según los patrones de su carga de trabajo, existe un límite de 1000 particiones por nodo para Elasticsearch y OpenSearch de la versión 1.1 a la 2.15 y de 4000 para OpenSearch 2.17 y versiones

posteriores. La API cat/allocation proporciona una vista rápida de la cantidad de particiones y el almacenamiento total de particiones en los nodos de datos.

Proporción de partición a CPU: cuando una partición participa en una solicitud de indexación o búsqueda, utiliza una vCPU para procesar la solicitud. Como práctica recomendada, utilice un punto de escala inicial de 1,5 vCPU por partición. Si su tipo de instancia tiene 8 vCPU, configure el recuento de nodos de datos para que cada nodo no tenga más de seis particiones. Tenga en cuenta que se trata de una aproximación. Asegúrese de probar la carga de trabajo y escalar el clúster en consecuencia.

Para obtener recomendaciones sobre el volumen de almacenamiento, el tamaño de las particiones y el tipo de instancia, consulte los recursos siguientes:

- the section called “Determinación del tamaño de dominios”
- the section called “Escala de petabytes”

Evite el sesgo de almacenamiento

El sesgo de almacenamiento ocurre cuando uno o más nodos de un clúster contienen una mayor proporción de almacenamiento para uno o más índices que los demás. Los indicios de que hay sesgo de almacenamiento incluyen utilización desigual de la CPU, latencia intermitente y desigual, y colocación en cola desigual en los nodos de datos. Para determinar si tiene problemas de sesgo, consulte las siguientes secciones de solución de problemas:

- the section called “Partición de nodos y sesgo de almacenamiento”
- the section called “Partición del índice y el sesgo de almacenamiento”

Stability

Las siguientes prácticas recomendadas se aplican para mantener un dominio de OpenSearch Service estable y saludable.

Manténgase al día con OpenSearch

Actualizaciones del software del servicio

OpenSearch Service publica de forma periódica actualizaciones del software del sistema que agregan nuevas características o mejoran sus dominios. Las actualizaciones no cambian la versión

del motor de OpenSearch o Elasticsearch. Se recomienda programar un tiempo recurrente para ejecutar la operación de la API DescribeDomain, y activar una actualización del software de servicio si `UpdateStatus` es `ELIGIBLE`. Si no actualiza el dominio en un determinado tiempo (por lo general, dos semanas), OpenSearch Service realiza la actualización automáticamente.

Actualizaciones de la versión de OpenSearch

OpenSearch Service agrega regularmente soporte para las versiones de OpenSearch mantenidas por la comunidad. Actualice siempre a las versiones más recientes de OpenSearch cuando estén disponibles.

OpenSearch Service actualiza simultáneamente los OpenSearch y OpenSearch Dashboards (o Elasticsearch y Kibana si su dominio ejecuta un motor heredado). Si el clúster tiene nodos maestros dedicados, las actualizaciones se completan sin tiempo de inactividad. De lo contrario, el clúster podría no responder durante varios segundos después de la actualización mientras elige un nodo maestro. OpenSearch Dashboards podría no estar disponible durante parte o la totalidad de la actualización.

Hay dos formas de actualizar un dominio:

- Actualización local: esta opción es más fácil porque mantiene el mismo clúster.
- Actualización de instantáneas y restauración: esta opción es buena para probar nuevas versiones en un clúster nuevo o migrar entre clústeres.

Independientemente del proceso de actualización que utilice, recomendamos mantener un dominio que sea únicamente para desarrollo y pruebas y actualizarlo a la nueva versión antes de actualizar el dominio de producción. Para el tipo de implementación, seleccione Desarrollo y pruebas cuando cree el dominio de prueba. Asegúrese de actualizar todos los clientes a versiones compatibles inmediatamente después de la actualización del dominio.

Cómo mejorar el rendimiento de las instantáneas

Para evitar que la instantánea se bloquee durante el procesamiento, el tipo de instancia del nodo maestro dedicado debe coincidir con el número de particiones. Para más información, consulte the section called “Elección de tipos de instancias para nodos principales dedicados”. Además, cada nodo no debe tener más de las 25 particiones recomendadas por cada GiB de memoria dinámica de Java. Para más información, consulte the section called “Selección del número de particiones”.

Habilite los nodos maestros dedicados

Nodos maestros dedicados para mejorar la estabilidad de los clústeres. Un nodo maestro dedicado realiza tareas de administración de clústeres, pero no contiene datos de índice ni responde a las solicitudes de los clientes. Al asumir de este modo las tareas de administración de clústeres, aumenta la estabilidad de su dominio y hace posible que se lleven a cabo algunos cambios de configuración sin tiempo de inactividad.

Habilite y utilice tres nodos maestros dedicados para lograr una estabilidad óptima del dominio en tres zonas de disponibilidad. La implementación con Multi-AZ con modo de espera le permite configurar tres nodos maestros dedicados. Para obtener recomendaciones de tipos de instancias, consulte the section called “Elección de tipos de instancias para nodos principales dedicados”.

Implemente en varias zonas de disponibilidad

Para evitar que se pierdan datos y minimizar el tiempo de inactividad del clúster en caso de que se produzca una interrupción del servicio, puede distribuir los nodos en dos o tres zonas de disponibilidad de la misma Región de AWS. La práctica recomendada es realizar la implementación mediante Multi-AZ con modo de espera, que configura tres zonas de disponibilidad: dos zonas activas y una en espera, y con dos particiones de réplicas por índice. Esta configuración permite a OpenSearch Service distribuir particiones de réplicas en zonas de disponibilidad diferentes de las de sus correspondientes particiones principales. No hay cargos por transferencia de datos entre zonas de disponibilidad para las comunicaciones de clústeres entre zonas de disponibilidad.

Las zonas de disponibilidad son ubicaciones aisladas dentro de cada región. Con una configuración de dos zonas de disponibilidad, perder una zona de disponibilidad significa que se pierde la mitad de toda la capacidad del dominio. El cambio a tres zonas de disponibilidad reduce aún más el impacto de perder una sola zona de disponibilidad.

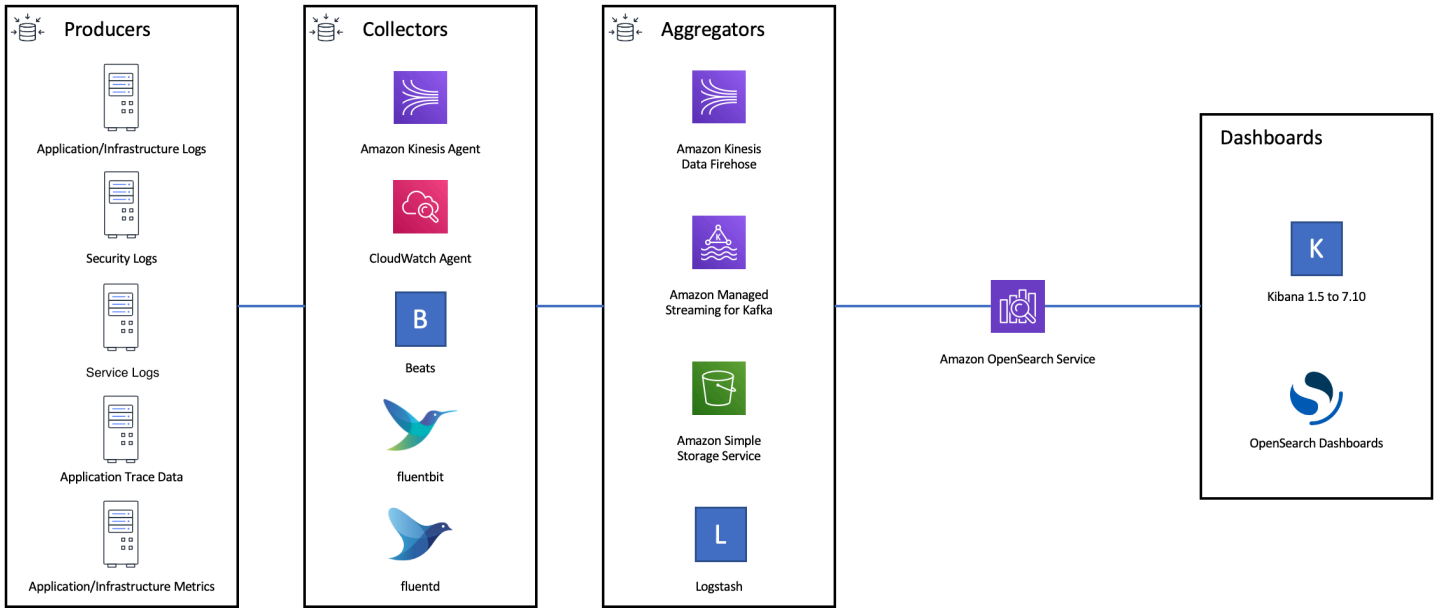
Controle el flujo de incorporación y el almacenamiento en búfer

Recomendamos que se limite el recuento total de solicitudes mediante la operación de la API _bulk. Es más eficiente enviar una solicitud `_bulk` que contiene 5000 documentos que enviar 5000 solicitudes que contienen un solo documento.

Para lograr una estabilidad operativa óptima, a veces es necesario limitar o incluso detener el flujo ascendente de las solicitudes de indexación. Limitar la tasa de solicitudes de índices es un mecanismo importante para hacer frente a picos inesperados u ocasionales en las solicitudes que,

de otro modo, podrían abrumar al clúster. Considere la posibilidad de crear un mecanismo de control de flujo en su arquitectura ascendente.

El siguiente diagrama muestra varias opciones de componentes para una arquitectura de incorporación de registros. Configure la capa de agregación para dejar espacio suficiente para almacenar en búfer los datos entrantes para picos de tráfico repentinos y un breve mantenimiento del dominio.



Cree asignaciones para cargas de trabajo de búsqueda

Para cargas de trabajo de búsqueda, cree asignaciones que definen cómo OpenSearch almacena e indexa los documentos y sus campos. Establezca `dynamic` en `strict` para evitar la adición accidental de nuevos campos.

```
PUT my-index
{
  "mappings": {
    "dynamic": "strict",
    "properties": {
      "title": { "type" : "text" },
      "author": { "type" : "integer" },
      "year": { "type" : "text" }
    }
  }
}
```

Utilice plantillas de índice

Puede utilizar una plantilla de índice como una forma de indicar a OpenSearch cómo configurar un índice cuando se crea. Configure las plantillas de índices antes de crear índices. Luego, cuando crea un índice, hereda la configuración y las asignaciones de la plantilla. Puede aplicar más de una plantilla a un único índice, de manera que pueda especificar la configuración en una plantilla y las asignaciones en otra. Esta estrategia permite una plantilla para configuraciones comunes en varios índices y plantillas separadas para configuraciones y asignaciones más específicas.

Los siguientes parámetros son útiles para configurar en plantillas:

- Número total de particiones primarias y de réplicas
- Intervalo de actualización (frecuencia con la que se actualizan y se realizan cambios recientes en el índice disponibles para la búsqueda)
- Control de mapeo dinámico
- Asignaciones de campo explícito

La siguiente plantilla de ejemplo contiene cada una de estas configuraciones:

```
{
   "index_patterns":[
      "index-*"
   ],
   "order": 0,
   "settings": {
      "index": {
         "number_of_shards": 3,
         "number_of_replicas": 1,
         "refresh_interval": "60s"
      }
   },
   "mappings": {
      "dynamic": false,
      "properties": {
         "field_name1": {
            "type": "keyword"
         }
      }
   }
}
```

Incluso si rara vez cambian, tener la configuración y las asignaciones definidas de forma centralizada en OpenSearch es más fácil de administrar que actualizar varios clientes ascendentes.

Administre los índices con la administración de estado de índice

Si administra registros o datos de serie temporal, le recomendamos que utilice Administración de estados de índice (ISM). ISM le permite automatizar las tareas regulares de administración del ciclo de vida. Con ISM, puede crear políticas que invoquen la renovación de alias de índice, tomen instantáneas de índices, muevan índices entre niveles de almacenamiento y eliminen índices antiguos. Incluso puede usar la operación sustitución de ISM como estrategia alternativa de administración del ciclo de vida de los datos para evitar el sesgo de las particiones.

En primer lugar, establezca una política de ISM. Por ejemplo, consulte the section called “Ejemplos de política”. A continuación, puede adjuntar la política a uno o más índices. Si incluye un campo Plantilla de ISM en la política, OpenSearch Service aplica automáticamente la política a cualquier índice que coincida con el patrón especificado.

Eliminar los índices que no se utilizan

Revise periódicamente los índices de su clúster e identifique los que no estén en uso. Tome una instantánea de esos índices para que se almacenen en S3 y, a continuación, elimínelos. Cuando se eliminan los índices no utilizados, se reduce el recuento de particiones y se permite una distribución del almacenamiento de información y una utilización de recursos más equilibradas entre los nodos. Incluso cuando están inactivos, los índices consumen algunos recursos durante las actividades de mantenimiento de índices internos.

En lugar de eliminar manualmente los índices no utilizados, puede usar ISM para tomar automáticamente una instantánea y eliminar los índices después de un periodo de tiempo determinado.

Utilice varios dominios para disfrutar de una alta disponibilidad

Para lograr una alta disponibilidad más allá del 99,9 % de tiempo de actividad en varias regiones, considere la posibilidad de usar dos dominios. Para conjuntos de datos pequeños o que cambian lentamente, puede configurar la replicación entre clústerespara mantener un modelo activo-pasivo. En este modelo, solo se escribe el dominio principal, pero se puede leer cualquiera de los dos dominios. Para conjuntos de datos más grandes y datos que cambian rápidamente, configure la entrega dual en la canalización de incorporación para que todos los datos se escriban de forma independiente en ambos dominios en un modelo activo-activo.

Diseñe sus aplicaciones ascendentes y descendentes teniendo en cuenta la conmutación por error. Asegúrese de probar el proceso de conmutación por error junto con otros procesos de recuperación de desastres.

Rendimiento

Las siguientes prácticas recomendadas se aplican para ajustar sus dominios para lograr un rendimiento óptimo.

Optimice el tamaño y la compresión de solicitudes

El tamaño masivo depende de los datos, el análisis y la configuración del clúster, pero un buen punto de partida es de 3 a 5 MiB por solicitud masiva.

Envíe solicitudes y reciba respuestas de sus dominios de OpenSearch mediante compresión gzip para reducir el tamaño de la carga útil de las solicitudes y respuestas. Puede usar la compresión gzip con el cliente Python de OpenSearch, o mediante la inclusión de los siguientes encabezados desde el lado del cliente:

- `'Accept-Encoding': 'gzip'`
- `'Content-Encoding': 'gzip'`

Para optimizar el tamaño de las solicitudes, comience con un tamaño de solicitud de 3 MiB. Luego, aumente lentamente el tamaño de la solicitud hasta que el desempeño de la indexación deje de mejorar.

> **Note**
>
> Para habilitar la compresión gzip en dominios que ejecutan Elasticsearch versión 6.x, debe establecer `http_compression.enabled` a nivel de clúster. Esta configuración se cumple de forma predeterminada en las versiones 7.x de Elasticsearch y en todas las versiones de OpenSearch.

Reduzca el tamaño de las respuestas de solicitudes masivas

Para reducir el tamaño de las respuestas de OpenSearch, excluya los campos innecesarios con el parámetro `filter_path`. Asegúrese de no filtrar ningún campo que sea necesario para identificar

o reintentar las solicitudes fallidas. Para obtener más información y ejemplos, consulta the section called “Reducción del tamaño de la respuesta”.

Ajuste intervalos de actualización

Los índices de OpenSearch al final tienen consistencia de lectura. Una operación de actualización hace que todas las actualizaciones que se realizan en un índice estén disponibles para la búsqueda. El intervalo de actualización predeterminado es de un segundo, lo que significa que OpenSearch realiza una actualización cada segundo mientras se escribe un índice.

Cuando menos frecuente sea la actualización de un índice (mayor intervalo de actualización), mejor será el rendimiento general de la indexación. La desventaja de aumentar el intervalo de actualización es que hay un retraso mayor entre la actualización del índice y el momento en que los nuevos datos están disponibles para la búsqueda. Establezca el intervalo de actualización tan alto como pueda tolerar para mejorar el rendimiento general.

Recomendamos configurar el parámetro `refresh_interval` para todos los índices a 30 segundos o más.

Habilite el ajuste automático

La función de ajuste automático utiliza métricas de rendimiento y uso de un clúster de OpenSearch para sugerir cambios de configuración tamaños de cola y caché y de máquina virtual Java (JVM) en los nodos. Estos cambios opcionales mejoran la velocidad y la estabilidad del clúster. Puede volver a la configuración predeterminada de OpenSearch Service en cualquier momento. La función de ajuste automático está habilitada de forma predeterminada en nuevos dominios, a menos que la desactive explícitamente.

Recomendamos que se habilite el ajuste automático en todos los dominios y se establezca un período de mantenimiento periódico o se revisen periódicamente sus recomendaciones.

Seguridad

Las siguientes prácticas recomendadas se aplican a la protección de sus dominios.

Cómo habilitar el control de acceso detallado

El control de acceso preciso le permite controlar quién puede acceder a ciertos datos dentro de un dominio de OpenSearch Service. En comparación con el control de acceso generalizado, el control de acceso detallado proporciona a cada clúster, índice, documento y campo su propia política de

acceso especificada. Los criterios de acceso pueden basarse en una serie de factores, incluido el rol de la persona quien solicita el acceso y la acción que pretende realizar con los datos. Por ejemplo, puede dar a un usuario acceso para escribir en un índice, y a otra se le puede dar acceso solo para leer los datos del índice sin realizar cambios.

El control de acceso detallado permite que los datos con diferentes requisitos de acceso existan en el mismo espacio de almacenamiento sin tener problemas de seguridad o cumplimiento.

Se recomienda habilitar un control de acceso detallado para sus dominios.

Implemente dominios dentro de una VPC

Al situar un dominio de OpenSearch Service dentro de una nube privada virtual (VPC), se consigue una comunicación segura entre OpenSearch Service y los demás servicios dentro de la VPC, sin necesidad de una puerta de enlace de Internet, un dispositivo NAT o una conexión de VPN. Todo el tráfico permanece seguro dentro de la nube de AWS. Debido a su aislamiento lógico, los dominios que residen dentro de una VPC tienen una capa adicional de seguridad en comparación con los dominios que utilizan puntos de conexión públicos.

Le recomendamos que utilice crear sus dominios dentro de una VPC.

Aplique una política de acceso restrictivo

Incluso si su dominio se implementa dentro de una VPC, es una práctica recomendada la implementación de la seguridad en capas. Asegúrese de comprobar la configuración de sus políticas de acceso actuales.

Aplique una política de acceso basado en recursos restrictiva en el dominio y siga el principio de privilegios mínimos para conceder acceso a la API de configuración y las operaciones de API de OpenSearch. Como regla general, evite usar la entidad principal de usuario anónimo `"Principal": {"AWS": "*" }` en sus políticas de acceso.

Sin embargo, hay algunas situaciones en las que es aceptable usar una política de acceso abierto, como cuando se habilita un control de acceso detallado. Una política de acceso abierto puede permitirle acceder al dominio en los casos en que la firma de solicitudes sea difícil o imposible, por ejemplo, desde ciertos clientes y herramientas.

Habilite el cifrado en reposo

Los dominios de OpenSearch Service ofrecen cifrado de datos en reposo para prevenir el acceso no autorizado a los datos. El cifrado en reposo utiliza AWS Key Management Service (AWS KMS) para

almacenar y administrar sus claves de cifrado y el algoritmo estándar de cifrado avanzado con claves de 256 bits (AES-256) para realizar el cifrado.

Si el dominio almacena información confidencial, habilite el cifrado de datos en reposo.

Habilite el cifrado de nodo a nodo

El cifrado de nodo a nodo proporciona una capa de seguridad adicional además de las características predeterminadas de seguridad dentro de OpenSearch Service. Implementa la seguridad de la capa de transporte (TLS) para todas las comunicaciones entre los nodos que están aprovisionadas en OpenSearch. El cifrado de nodo a nodo garantiza que todos los datos enviados a su dominio de OpenSearch Service a través de HTTPS permanezcan cifrados en tránsito mientras se distribuyen y replican entre nodos.

Si el dominio almacena información confidencial, habilite el cifrado de nodo a nodo.

Monitorización de con AWS Security Hub CSPM

Monitoree el uso de OpenSearch Service en relación con las prácticas recomendadas de seguridad con AWS Security Hub CSPM. Security Hub utiliza controles de seguridad para evaluar las configuraciones de los recursos y los estándares de seguridad para ayudarle a cumplir varios marcos de conformidad. Para obtener más información sobre el uso de Security Hub para evaluar los recursos de OpenSearch Service, consulte Controles de Amazon OpenSearch Service en la Guía del usuario de AWS Security Hub CSPM.

Optimización de costos

Las siguientes prácticas recomendadas se aplican a la optimización y el ahorro en los costos de OpenSearch Service.

Use los tipos de instancia de última generación

OpenSearch Service siempre está adoptando nuevos tipos de instancias de Amazon EC2 que ofrecen un mejor rendimiento a un costo menor. Recomendamos usar siempre las instancias de última generación.

Evite usar instancias T2 o `t3.small` para dominios de producción, ya que pueden volverse inestables bajo una carga pesada sostenida. Las instancias `r6g.large` son una opción para cargas de trabajo de producción pequeñas (tanto como nodos de datos, como nodos maestros dedicados).

Use los volúmenes gp3 de Amazon EBS más recientes

Los nodos de datos de OpenSearch requieren un almacenamiento de baja latencia y alto rendimiento para proporcionar una indexación y una consulta rápidas. Utilizando volúmenes gp3 de Amazon EBS, se obtiene un mayor rendimiento de referencia (IOPS y rendimiento) a un costo un 9,6 % inferior que con el tipo de volumen gp2 de Amazon EBS que se ofrecía anteriormente. Puede aprovisionar IOPS y rendimiento adicionales independientemente del tamaño de los volúmenes mediante gp3. Además, estos volúmenes son más estables que los de la generación anterior, ya que no utilizan créditos de ráfaga. Asimismo, el tipo de volumen gp3 duplica los límites de tamaño de volumen por nodo de datos en comparación con el tipo de volumen gp2. Con estos volúmenes de mayor tamaño, puede reducir el costo de los datos pasivos mediante el incremento de la cantidad de almacenamiento por nodo de datos.

Utilice almacenamiento en frío y UltraWarm para datos de registro de series temporales

Si usa OpenSearch para el análisis de registros, mueva sus datos a UltraWarm o almacenamiento en frío para reducir los costos. Utilice la Administración de estado de índices (ISM) para migrar datos entre niveles de almacenamiento y administrar la retención de datos.

UltraWarm proporciona una forma rentable de almacenar grandes cantidades de datos de solo lectura en OpenSearch Service. UltraWarm utiliza Amazon S3 para el almacenamiento, lo que significa que los datos son inmutables y solo se necesita una copia. Solo paga por un almacenamiento que es equivalente al tamaño de las particiones principales de sus índices. Las latencias de las consultas UltraWarm aumentan con la cantidad de datos de S3 que son necesarios para atender la consulta. Después de que los datos se han almacenado en caché en los nodos, las consultas a los índices UltraWarm funcionan de manera similar a las consultas a los índices activos.

El almacenamiento en frío también cuenta con el respaldo de S3. Cuando necesite consultar datos almacenados en frío, puede adjuntarlos de forma selectiva a los nodos UltraWarm existentes. Los datos inactivos incurren en el mismo costo de almacenamiento administrado que UltraWarm, pero los objetos en el almacenamiento en frío no consumen recursos de nodo UltraWarm. Por lo tanto, el almacenamiento en frío proporciona una cantidad significativa de capacidad de almacenamiento sin afectar el tamaño o el recuento de nodos UltraWarm.

UltraWarm se vuelve rentable cuando se tienen aproximadamente 2,5 TiB de datos para migrar desde el almacenamiento en caliente. Supervise su tasa de llenado y planifique mover los índices a UltraWarm antes de alcanzar ese volumen de datos.

Revise las recomendaciones para instancias reservadas

Considere comprar instancias reservadas (RI) después de contar con una buena referencia sobre el rendimiento y el consumo de cómputos. Los descuentos comienzan en torno al 30 % para reservas sin pago anticipado de 1 año y pueden aumentar hasta un 50 % para todos los compromisos de pagos iniciales de 3 años.

Después de observar un funcionamiento estable durante al menos 14 días, revise Acceder a las recomendaciones de reservas en la Guía del usuario de AWS Cost Management. El encabezado de Amazon OpenSearch Service muestra recomendaciones de compra de instancias reservadas específicas y ahorros proyectados.

Tamaño de los dominios de Amazon OpenSearch Service

No existe un método perfecto para determinar el tamaño de dominios de Amazon OpenSearch Service. Sin embargo, para comenzar a comprender las necesidades de almacenamiento, el servicio y el propio OpenSearch, puede realizar una estimación inicial fundamentada de las necesidades de equipo. Este cálculo puede servir como un buen punto de partida para la mayoría de los aspectos esenciales de la determinación del tamaño de los dominios: probarlos con cargas de trabajo representativas y monitorear su rendimiento.

Temas

- Cálculo de requisitos de almacenamiento
- Selección del número de particiones
- Selección de tipos de instancias y pruebas

Cálculo de requisitos de almacenamiento

La mayoría de las cargas de trabajo de OpenSearch se dividen en dos amplias categorías principales:

- Índice de larga duración: escribe un código que procesa los datos en uno o varios índices de OpenSearch y, luego, actualiza esos índices periódicamente a medida que cambia los datos de origen. Algunos ejemplos comunes son las búsquedas en sitios web, documentos y comercio electrónico.
- Índices continuos: los datos fluyen de manera continua en un conjunto de índices temporales, con un periodo de indexación y de retención (como un conjunto de índices diarios que se conservan

durante dos semanas). Algunos ejemplos comunes son el análisis de registros, el procesamiento de series temporales y el análisis de secuencias de clics.

Para las cargas de trabajo de índices de larga duración, puede examinar el origen de datos en disco y determinar fácilmente la cantidad de espacio de almacenamiento que consumen. Si los datos provienen de varias fuentes, solo tiene que agregar esas fuentes juntas.

Para los índices continuos, puede multiplicar la cantidad de datos generados durante un periodo de tiempo representativo por el periodo de retención. Por ejemplo, si genera 200 MiB de datos de registro por hora, o sea 4,7 GiB por día, eso significa que contará con un total de 66 GiB de datos en un momento dado si hubo un periodo de retención de dos semanas.

Sin embargo, el tamaño de los datos de origen es solo uno de los aspectos de las necesidades de almacenamiento. También debe considerar lo siguiente:

- Número de réplicas: cada réplica es una copia completa de la partición principal; el tamaño de almacén del índice muestra el tamaño que ocupan la partición principal y de réplica. De forma predeterminada, cada índice de OpenSearch posee una sola réplica. Recomendamos tener al menos una para evitar la pérdida de datos. Las réplicas también mejoran el rendimiento, por lo que tal vez convenga tener más si tiene una carga de trabajo que realiza muchas operaciones de lectura. Utilice `PUT /my-index/_settings` para actualizar la configuración `number_of_replicas` para su índice.
- Sobrecarga de indexación de OpenSearch: el tamaño en disco de un índice varía. El tamaño total de los datos de origen más el índice suele ser del 110 % del origen, y el índice de hasta el 10 % de los datos de origen. Después de indexar los datos, puede utilizar la API `_cat/indices?v` y el valor de `pri.store.size` para calcular la sobrecarga exacta. `_cat/allocation?v` también proporciona un resumen útil.
- Espacio reservado por el sistema operativo: de forma predeterminada, Linux reserva el 5 % del sistema de archivos para el usuario `root` para procesos críticos, recuperación del sistema y como medida de seguridad frente a problemas de fragmentación del disco.
- Sobrecarga de OpenSearch Service: OpenSearch Service reserva el 20 % del espacio de almacenamiento de cada instancia (hasta un máximo de 20 GiB) para combinaciones de segmentos, registros y otras operaciones internas.

  Debido a este máximo de 20 GiB, la cantidad total de espacio reservado puede variar drásticamente en función del número de instancias del dominio. Por ejemplo, un dominio podría tener tres instancias `m6g.xlarge.search`, cada una con 500 GiB de espacio de

almacenamiento, lo que equivaldría a un total de 1,46 TiB. En este caso, el espacio reservado total solo es de 60 GiB. Otro dominio podría tener 10 instancias `m3.medium.search`, cada una con 100 GiB de espacio de almacenamiento, lo que equivaldría a un total de 0,98 TiB. Aquí, el espacio reservado total es de 200 GiB, aunque el primer dominio sea un 50 % mayor.

En la siguiente fórmula, aplicamos una estimación “en el peor de los casos” de los gastos generales. Este cálculo incluye espacio libre adicional para ayudar a minimizar el impacto de los errores de nodo y las interrupciones en la zona de disponibilidad.

En resumen, si tiene 66 GiB de datos en un momento dado y quiere una réplica, el requisito de almacenamiento mínimo será aproximadamente 66 * 2 * 1,1 / 0,95 / 0,8 = 191 GiB. Se podría generalizar este cálculo de la manera siguiente:

Datos de origen * (1 + número de réplicas) * (1 + sobrecarga de indexación) / (1 - espacio reservado de Linux) / (1 - sobrecarga de OpenSearch Service) = requisito mínimo de almacenamiento

O puede utilizar esta versión simplificada:

Datos de origen * (1 + número de réplicas) * 1,45 = requisito mínimo de almacenamiento

La falta de espacio de almacenamiento suficiente es una de las causas más comunes de la inestabilidad del clúster. Por lo tanto, debe verificar los números cuando elige tipos de instancia, recuentos de instancias y volúmenes de almacenamiento.

Existen otras consideraciones respecto al almacenamiento:

- Si el requisito mínimo de almacenamiento es superior a 1 PB, consulte the section called “Escala de petabytes”.
- Si tiene índices continuos y desea usar una arquitectura caliente/templada, consulte the section called “Almacenamiento UltraWarm”.

Selección del número de particiones

Una vez que conozca los requisitos de almacenamiento, puede investigar la estrategia de indexación. De forma predeterminada, en OpenSearch Service cada índice se divide en cinco particiones principales y una réplica (en total 10 particiones). Este comportamiento difiere del de OpenSearch de código abierto, que utiliza de forma predeterminada una partición principal y una de réplica. Como no se puede cambiar fácilmente el número de particiones principales de un índice existente, debe decidir el número de particiones antes de indexar el primer documento.

El objetivo general de la selección de un número de particiones es distribuir un índice de manera uniforme entre todos los nodos de datos del clúster. Sin embargo, estas particiones no deberían ser demasiado grandes ni demasiado numerosas. Una pauta general es tratar de mantener el tamaño de la partición entre 10 y 30 GiB para las cargas de trabajo en las que la latencia de búsqueda es un objetivo de rendimiento clave y entre 30 y 50 GiB para las cargas de trabajo con mucha escritura, como el análisis de registros.

Las particiones grandes pueden dificultar a OpenSearch recuperarse de un error, pero como cada partición utiliza una cantidad de CPU y memoria, tener demasiadas particiones pequeñas podría provocar problemas de rendimiento y errores por memoria insuficiente. En otras palabras, las particiones deben ser lo suficientemente pequeñas para que la instancia de OpenSearch Service subyacente pueda administrarlas, pero no tan pequeñas que requieran un esfuerzo inútil por parte del equipo.

Por ejemplo, supongamos que tiene 66 GiB de datos. No espera que el número aumente a lo largo del tiempo y desea mantener las particiones en torno a los 30 GiB cada una. El número de particiones debería ser, por tanto, aproximadamente 66 * 1,1/30 = 3. Se podría generalizar este cálculo de la manera siguiente:

(Datos de origen + espacio para crecer) * (1 + sobrecarga de indexación) / tamaño de partición deseado = número aproximado de particiones principales

Esta ecuación ayuda a compensar el aumento de los datos a lo largo del tiempo. Si prevé que estos mismos 66 GiB de datos se cuadrupliquen a lo largo del próximo año, el número aproximado de particiones es (66 + 198) * 1,1/30 = 10. Recuerde, sin embargo, que aún no tiene esos 198 GiB de datos adicionales. Asegúrese de que estos preparativos para el futuro no creen particiones innecesariamente diminutas que consuman grandes cantidades de CPU y memoria en la actualidad. En este caso, tendrá 66 * 1,1/10 particiones = 7,26 GiB por partición, lo que consumirá recursos adicionales y queda debajo del intervalo de tamaño recomendado. Podría considerar el enfoque más intermedio de seis particiones, lo que dejaría particiones de 12 GiB hoy y particiones de 48 GiB en el futuro. De nuevo, es posible que prefiera empezar con tres particiones y reindexar sus datos cuando las particiones superen los 50 GiB.

Un problema mucho menos común implica limitar el número de particiones por nodo. Si el tamaño de las particiones es adecuado, normalmente se queda sin espacio en disco mucho antes de alcanzar este límite. Por ejemplo, una instancia de `m6g.large.search` tiene un tamaño máximo de disco de 512 GiB. Si permanece por debajo del 80 % de uso del disco y el tamaño de sus particiones en 20 GiB, puede acomodar aproximadamente 20 particiones. Elasticsearch 7.x y posteriores y todas las versiones de OpenSearch hasta la 2.15, tienen un límite de 1000

particiones por nodo. Para ajustar el máximo de particiones por nodo, establezca la configuración `cluster.max_shards_per_node`. Para OpenSearch 2.17 y versiones posteriores, OpenSearch Service admite 1000 particiones por cada 16 GB de memoria de heap de la JVM, hasta un máximo de 4000 particiones por nodo. Para ver un ejemplo, consulte Configuración del clúster. Para obtener más información sobre la cantidad de particiones, consulte the section called “Cuotas de recuento de la partición”.

El dimensionamiento apropiado de las particiones casi siempre lo mantiene por debajo de este límite, pero también puede considerar el número de particiones por cada GiB del montón de Java. En un nodo dado, no tenga más de 25 particiones por cada GiB del montón de Java. Por ejemplo, una instancia de `m5.large.search` tiene un montón de 4 GiB, por lo que cada nodo no debe tener más de 100 particiones. En ese recuento de particiones, cada uno tiene aproximadamente 5 GiB de tamaño, lo que está muy por debajo de nuestra recomendación.

Selección de tipos de instancias y pruebas

Después de calcular sus requisitos de almacenamiento y elegir el número de particiones que necesita, puede comenzar a tomar decisiones sobre el equipo. Los requisitos del equipo varían enormemente en función de la carga de trabajo, pero aquí también ofrecemos algunas recomendaciones básicas.

En general, los límites de almacenamiento para cada tipo de instancia se corresponden con la cantidad de CPU y memoria que podría necesitar para las cargas de trabajo. Por ejemplo, una instancia `m6g.large.search` tiene un tamaño de volumen de EBS máximo de 512 GiB, 2 núcleos de vCPU y 8 GiB de memoria. Si el clúster tiene muchas particiones, realiza un gran número de altas, actualiza documentos frecuentemente o procesa un gran número de consultas, esos recursos podrían ser insuficientes para sus necesidades. Si su clúster entra dentro de una de estas categorías, empiece con una configuración más cercana a 2 núcleos de vCPU y 8 GiB de memoria para cada 100 GiB de su requisito de almacenamiento.

> **Tip**
>
> Para obtener un resumen de los recursos del equipo que se asignan a cada tipo de instancia, consulte Precio de Amazon OpenSearch Service.

Aun así, estos recursos podrían ser insuficientes. Algunos usuarios de OpenSearch informan que necesitan muchos más de estos recursos para satisfacer sus necesidades. Para encontrar el equipo

correcto para la carga de trabajo, debe realizar una estimación inicial fundamentada, probarla con cargas de trabajo representativas, ajustarla y probarla de nuevo.

Paso 1: realizar una estimación inicial

Para comenzar, recomendamos elegir tres nodos como mínimo para evitar posibles problemas de OpenSearch, como el estado de cerebro dividido (cuando un lapso de comunicación conduce a que un clúster tenga dos nodos maestros). Aunque tenga tres nodos maestros dedicados, seguiremos recomendando que tenga un mínimo de dos nodos de datos para la replicación.

Paso 2: calcular los requisitos de almacenamiento por nodo

Si tuviera un requisito de almacenamiento de 184 GiB y el número mínimo recomendado de tres nodos, usaría la ecuación 184 / 3 = 61 GiB para encontrar la cantidad de almacenamiento que necesita cada nodo. En este ejemplo, podría elegir tres instancias `m6g.large.search`, donde cada una con un volumen de almacenamiento de EBS de 90 GiB para no quedarse corto y tener un margen de crecimiento para el futuro. Esta configuración proporciona 6 núcleos de vCPU y 24 GiB de memoria, por lo que resulta adecuada para las cargas de trabajo más ligeras.

Para ver un ejemplo más sustancial, considere un requisito de almacenamiento de 14 TiB (14 336 GiB) y una carga de trabajo pesada. En este caso, podría decidir empezar con 2 * 144 = 288 núcleos de vCPU y 8 * 144 = 1 152 GiB de memoria. Estos números se corresponden con aproximadamente 18 instancias `i3.4xlarge.search`. Si no necesita el almacenamiento local rápido, también puede probar con 18 instancias `r6g.4xlarge.search`, cada una con un volumen de almacenamiento de EBS de 1 TiB.

Si el clúster incluye cientos de terabytes de datos, consulte the section called “Escala de petabytes”.

Paso 3: realizar pruebas representativas

Después de configurar el clúster, puede agregar sus índices con la ayuda del número de particiones calculado anteriormente, realizar algunas pruebas representativas en el cliente mediante un conjunto de datos realista y monitorear las métricas de CloudWatch para ver cómo el clúster se encarga de la carga de trabajo.

Paso 4: suceder o iterar

Si el rendimiento satisface sus necesidades, se superan las pruebas y las métricas de CloudWatch son normales, el clúster está listo para ser utilizado. Recuerde configurar alarmas de CloudWatch para detectar la utilización de recursos en mal estado.

Si el desempeño no es aceptable, no se superan las pruebas o los valores de `CPUUtilization` o `JVMMemoryPressure` son altos, es posible que tenga que elegir un tipo de instancia diferente (o agregar instancias) y continuar con las pruebas. Cuando se agregan instancias, OpenSearch reequilibra automáticamente la distribución de particiones en todo el clúster.

Debido a que es más fácil medir el exceso de capacidad en un clúster sobrealimentado que el déficit en uno infraalimentado, recomendamos comenzar con un clúster más grande de lo que crea necesario. A continuación, debe realizar pruebas y reducir verticalmente el tamaño hasta tener un clúster eficiente con los recursos adicionales precisos para garantizar operaciones estables durante los períodos de mayor actividad.

Los clústeres en producción o los clústeres con estados complejos se benefician de los nodos maestros dedicados, que mejoran el desempeño y la fiabilidad del clúster.

Escala de petabytes en Amazon OpenSearch Service

Los dominios de Amazon OpenSearch Service ofrecen almacenamiento adjunto de hasta 10 PB. Puede configurar un dominio con 1000 tipos de instancias `OR1.16xlarge.search`, cada una con 36 TB de almacenamiento. Debido a la gran diferencia de escala, las recomendaciones para los dominios de este tamaño difieren de nuestras recomendaciones generales. En esta sección, se explican consideraciones sobre la creación de dominios, los costos, el almacenamiento y el tamaño de las particiones.

Aunque esta sección hace referencia con frecuencia a los tipos de instancias `i3.16xlarge.search`, puede usar otros tipos de instancias para llegar a 10 PB de almacenamiento total del dominio.

Creación de dominios

> Los dominios de este tamaño superan el límite predeterminado de 80 instancias por dominio. Para solicitar un aumento del límite del servicio de hasta 1000 instancias por dominio, abra un caso en el Centro de asistencia de AWS.

Precios

> Antes de crear un dominio de este tamaño, consulte la página Precios de Amazon OpenSearch Service para asegurarse de que los costos asociados se ajusten a sus expectativas. Examine the section called “Almacenamiento UltraWarm” para ver si una arquitectura caliente/templada se ajusta a su caso de uso.

Almacenamiento

Los tipos de instancias `i3` se han diseñado para proporcionar almacenamiento rápido, local y de memoria no volátil (NVMe). Dado que este almacenamiento local suele ofrecer beneficios de rendimiento en comparación con Amazon Elastic Block Store, los volúmenes de EBS no son una opción cuando se seleccionan estos tipos de instancia en OpenSearch Service. Si prefiere el almacenamiento de EBS, utilice otro tipo de instancia, como `r6.12xlarge.search`.

Tamaño y número de fragmentos

Una directriz común de OpenSearch es no superar los 50 GB por partición. Habida cuenta del número de fragmentos necesarios para acomodar dominios grandes y los recursos disponibles para instancias `i3.16xlarge.search`, le recomendamos un tamaño de fragmento de 100 GB.

Por ejemplo, si tiene 450 TB de datos de origen y solo quiere una réplica, su requisito de almacenamiento mínimo será aproximadamente 450 TB * 2 * 1,1/0.95 = 1.04 PB. Para obtener una explicación de este cálculo, consulte the section called “Cálculo de requisitos de almacenamiento”. Aunque 1,04 PB / 15 TB = 70 instancias, podría seleccionar 90 o más instancias `i3.16xlarge.search` para disponer de una red de seguridad de almacenamiento, hacer frente a los errores de los nodos y tener en cuenta cierta variación en la cantidad de datos a lo largo del tiempo. Cada instancia agrega otros 20 GiB a su requisito de almacenamiento mínimo, pero para discos de este tamaño, esos 20 GiB son casi insignificantes.

Controlar el número de particiones no es tarea fácil. Los usuarios de OpenSearch suelen rotar los índices a diario y retener los datos durante una o dos semanas. En esta situación, puede que le resulte útil distinguir entre fragmentos "activos" e "inactivos". Los fragmentos activos son fragmentos que se están escribiendo o leyendo activamente. Las particiones inactivas pueden dar servicio a algunas solicitudes de lectura, pero están en gran medida inactivos. En general, debe mantener el número de fragmentos activos por debajo de varios miles. Cuando el número de particiones activas se acerca a los 10 000, surgen riesgos importantes de rendimiento y estabilidad.

Para calcular el número de fragmentos principales, utilice esta fórmula: 450 000 GB * 1,1 / 100 GB por fragmento = 4 950 fragmentos. Si se duplica ese número para dar cuenta de las réplicas se obtienen 9 900 particiones, lo que supone un grave problema si todas las particiones están activas. Pero si rota los índices y solo una séptima parte o una catorceava parte de las particiones están activas en un día determinado (1,414 o 707 particiones, respectivamente), el clúster puede funcionar correctamente. Como siempre, el paso más importante para determinar el tamaño y la configuración del dominio es realizar pruebas representativas en el cliente utilizando un conjunto de datos realista.

Nodos coordinadores dedicados en Amazon OpenSearch Service

Los nodos coordinadores dedicados de Amazon OpenSearch Service son nodos especializados que descargan las tareas de coordinación de los nodos de datos. Estas tareas incluyen la administración de las solicitudes de búsqueda y el alojamiento de OpenSearch Dashboards. Al separar estas funciones, los nodos coordinadores dedicados reducen la carga de los nodos de datos, lo que les permite centrarse en las operaciones de almacenamiento, indexación y búsqueda de datos. Esto mejora el rendimiento general del clúster y la utilización de los recursos.

Además, los nodos coordinadores dedicados ayudan a reducir la cantidad de direcciones IP privadas necesarias para las configuraciones de VPC, lo que conduce a una administración de red más eficiente. Esta configuración puede suponer una mejora de hasta un 15 % en el rendimiento de indexación y un 20 % en el rendimiento de las consultas, según las características de la carga de trabajo.

Cuándo utilizar nodos coordinadores dedicados

Los nodos coordinadores dedicados son los más beneficiosos en los siguientes escenarios.

- Clústeres grandes: en entornos con un gran volumen de datos o consultas complejas, delegar las tareas de coordinación a nodos dedicados puede mejorar el rendimiento del clúster.
- Consultas frecuentes: las cargas de trabajo que implican consultas de búsqueda o agregaciones frecuentes, especialmente aquellas con histogramas de fechas complejos o múltiples agregaciones, se benefician de un procesamiento de consultas más rápido.
- Uso intensivo de Dashboards: OpenSearch Dashboards pueden consumir muchos recursos. Al delegar esta responsabilidad en nodos coordinadores dedicados se reduce la carga sobre los nodos de datos.

Arquitectura y comportamiento

En un clúster de OpenSearch, los nodos coordinadores dedicados se ocupan de dos responsabilidades clave.

- Administración de solicitudes: estos nodos reciben las solicitudes de búsqueda entrantes y las reenvían a los nodos de datos correspondientes, que almacenan los datos relevantes. A continuación, consolidan los resultados de varios nodos de datos en un único conjunto de resultados globales, que se devuelve al cliente.

- Alojamiento de paneles: los nodos coordinadores administran OpenSearch Dashboards, lo que alivia a los nodos de datos de la carga adicional que supone alojar los OpenSearch Dashboards y administrar el tráfico relacionado.

En los dominios de VPC, a los nodos coordinadores dedicados se les asignan interfaces de red elásticas (ENI) en lugar de nodos de datos. Esta disposición ayuda a reducir la cantidad de direcciones IP privadas necesarias para las VPC, lo que mejora la eficiencia de la red. Por lo general, los nodos coordinadores dedicados representan aproximadamente el 10 % del total de nodos de datos.

Requisitos y limitaciones

Los nodos coordinadores dedicados tienen los siguientes requisitos y limitaciones.

- Los nodos coordinadores dedicados son compatibles con todas las versiones de OpenSearch y con las versiones de Elasticsearch de la 6.8 a la 7.10.
- Para habilitar los nodos coordinadores dedicados, el dominio debe tener habilitados los nodos maestros dedicados. Para obtener más información, consulte the section called “Nodos maestros dedicados”.
- El aprovisionamiento de nodos coordinadores dedicados puede dar lugar costos adicionales. Sin embargo, la mayor eficiencia de los recursos y el mejor rendimiento justifican la inversión, especialmente en el caso de clústeres grandes o complejos.

Aprovisionamiento de nodos coordinadores dedicados

Siga estos pasos para aprovisionar nodos coordinadores dedicados en un dominio existente. Asegúrese de que su dominio tenga habilitados los nodos maestros dedicados antes de aprovisionar los nodos coordinadores.

Consola

Para aprovisionar nodos coordinadores dedicados en la Consola de administración de AWS

1. Inicie sesión en la consola de Amazon OpenSearch Service en https://console.aws.amazon.com/aos/home.
2. Elija Dominios y, a continuación, seleccione el dominio que desea modificar.
3. En la sección Configuración de clúster, elija Editar.

4. Seleccione Activar nodos coordinadores dedicados.
5. Seleccione el tipo de instancia y el número de nodos coordinadores que aprovisionar.
6. Elija Guardar cambios. Puede que el dominio tarde varios minutos en actualizarse.

AWS CLI

Para aprovisionar nodos coordinadores dedicados mediante la AWS CLI, utilice el comando update-domain-config. El siguiente ejemplo aprovisiona tres nodos coordinadores `r6g.large.search` en un dominio.

```
aws opensearch update-domain-config \
  --domain-name my-opensearch-domain \
  --cluster-config
 InstanceCount=3,InstanceType=r6g.large.search,DedicatedCoordinatorCount=3,ZoneAwarenessEnabled
```

Este comando habilita los nodos coordinadores dedicados, establece el tipo de instancia y el recuento de los nodos coordinadores y habilita el reconocimiento de zona para una mayor disponibilidad.

Prácticas recomendadas

Tenga en cuenta las siguientes prácticas recomendadas al utilizar nodos coordinadores dedicados.

- Utilice instancias de uso general para la mayoría de los casos de uso. Proporcionan un enfoque equilibrado entre costo y rendimiento. Las instancias optimizadas para memoria son ideales para cargas de trabajo que requieren recursos de memoria sustanciales, como las que implican agregaciones complejas o búsquedas a gran escala.
- Un buen punto de partida es aprovisionar del 5 % al 10 % como nodos coordinadores dedicados. Por ejemplo, si su dominio tiene 90 nodos de datos de un tipo de instancia concreto, considere la posibilidad de aprovisionar de 5 a 9 nodos coordinadores del mismo tipo de instancia.

> **Note**
>
> La disponibilidad de los tipos de instancias puede variar según la región. Al seleccionar los tipos de instancia para los nodos coordinadores, compruebe que el tipo de instancia elegido esté disponible en la región de destino. Puede comprobar la disponibilidad del tipo de instancia en la consola de OpenSearch Service al crear o modificar su dominio.

- Para minimizar el riesgo de que se produzca un único punto de fallo, aprovisione al menos dos nodos coordinadores dedicados. Esto garantiza que el clúster permanezca operativo incluso si falla un nodo.
- Si usted utiliza búsqueda entre Regiones, aprovisione nodos coordinadores dedicados en los dominios de destino. Los dominios de origen por lo general no manejan tareas de coordinación.
- Para entornos con un uso intensivo de indexación, considere la posibilidad de utilizar instancias optimizadas para la CPU que coincidan con el tamaño de las instancias de sus nodos de datos para obtener un rendimiento óptimo.
- Para cargas de trabajo con uso intensivo de memoria, utilice un tipo de instancia ligeramente más grande para sus nodos coordinadores dedicados, a fin de ayudar a administrar el aumento en las demandas de memoria.
- Realice un seguimiento la métrica `CoordinatorCPUUtilization` de Amazon CloudWatch. Si supera constantemente el 80 %, podría indicar que necesita nodos coordinadores más grandes o adicionales para administrar la carga.
- Ajuste el tamaño de los nodos coordinadores dedicados para que coincidan con los nodos de datos. Por ejemplo, comience con 4 nodos coordinadores grandes de uso general cuando utilice 4 nodos de datos grandes.
- Utilice varias instancias más pequeñas en lugar de menos instancias más grandes para los nodos coordinadores, a menos que sus solicitudes o respuestas individuales necesiten una memoria extremadamente alta (en GB). Por ejemplo, elija 12 instancias 4xl en lugar de 6 instancias generales 8xlarge.

Recomendaciones de nodos por tamaño de clúster

Utilice las siguientes pautas como punto de partida para aprovisionar nodos coordinadores dedicados en función del tamaño del clúster. Ajusta la cantidad y el tipo de nodos en función de las características de la carga de trabajo y las métricas de rendimiento.

Tamaño del clúster	Nodos coordinadores recomendados	Tipo de instancia
Pequeño (hasta 50 nodos)	de 3 a 5 nodos	Fin general

Tamaño del clúster	Nodos coordinadores recomendados	Tipo de instancia
Medio (de 50 a 100 nodos)	de 5 a 9 nodos	Optimizada para memoria
Grande (más de 100 nodos)	10 a 15 nodos	Optimizada para memoria

Nodos maestros dedicados en Amazon OpenSearch Service

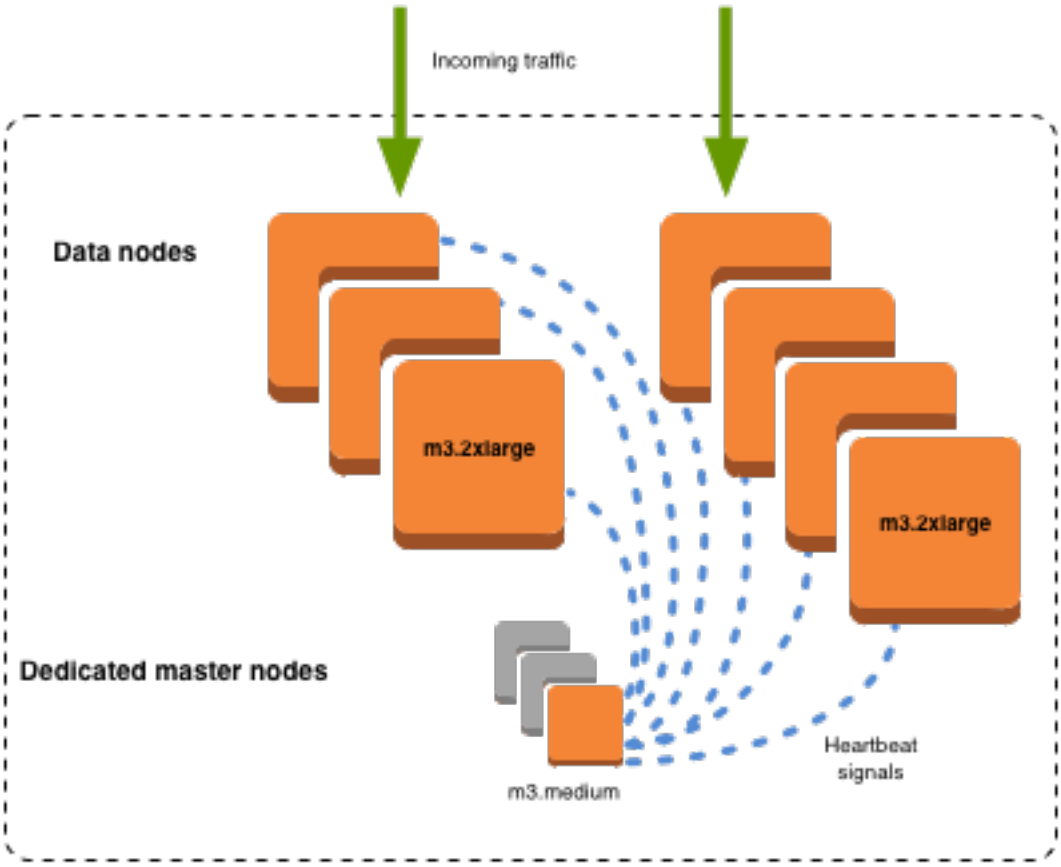
Amazon OpenSearch Service utiliza nodos maestros dedicados para aumentar la estabilidad de los clústeres. Un nodo maestro dedicado realiza tareas de administración de clústeres, pero que no contiene datos ni responde a las solicitudes de carga de datos. Al asumir de este modo las tareas de administración del clúster, aumenta la estabilidad del dominio. Al igual que todos los demás tipos de nodos, se paga una tarifa por hora por cada nodo maestro dedicado.

Los nodos maestros dedicados se encargan de realizar las siguientes tareas de administración del clúster:

- Seguimiento de todos los nodos del clúster.
- Seguimiento del número de índices del clúster.
- Seguimiento del número de particiones que pertenecen a cada índice.
- Mantenimiento de la información de enrutamiento para los nodos del clúster.
- Actualización del estado del clúster después de los cambios, como la creación de un índice o la adición o eliminación de nodos en el clúster.
- Reproducción de los cambios de estado del clúster en todos los nodos que lo componen.
- Monitorización del estado de todos los nodos del clúster, mediante el envío de señales de latido, señales periódicas que monitorizan la disponibilidad de los nodos de datos del clúster.

En la siguiente ilustración se muestra un dominio de OpenSearch Service con 10 instancias. Siete de ellas son nodos de datos y tres son nodos maestros dedicados. Solo uno de los nodos maestros dedicados está activo. Los dos nodos maestros dedicados grises están en reserva a la espera por si el nodo maestro dedicado activo sufre algún error. Los siete nodos de datos se encargan de atender

todas las solicitudes de carga de datos, mientras que el nodo maestro dedicado activo asume las tareas de administración del clúster.



Elección del número de nodos maestros dedicados

Recomendamos que utilice Multi-AZ con modo de espera, que agrega tres nodos maestros dedicados a cada dominio de producción de OpenSearch Service. Si realiza la implementación con Multi-AZ sin modo de espera o con Single-AZ, igual le recomendamos tres nodos maestros dedicados. No elija nunca un número par de nodos maestros dedicados. Tenga en cuenta lo siguiente al elegir el número de nodos maestros dedicados:

- Un nodo maestro dedicado está explícitamente prohibido por OpenSearch Service, ya que no dispone de copias de seguridad si se produce algún error. Recibirá una excepción de validación si intenta crear un dominio con un solo nodo maestro dedicado.
- Si tiene dos nodos maestros dedicados, el clúster no tiene el quórum de nodos necesario para elegir un nuevo nodo maestro si se produce algún error.

El quórum se calcula como el número de nodos maestros dedicados / 2 + 1 (redondeado al número entero más próximo). En este caso, 2 / 2 + 1 = 2. Como un nodo maestro dedicado produjo un error y solo existe una copia de seguridad, el clúster no tiene quórum y no puede elegir un nuevo nodo principal.

- El uso de tres nodos maestros dedicados, el número recomendado, ofrece dos nodos de backup en caso de que se produzca un error en un nodo maestro y el quórum necesario (2) para elegir un nuevo nodo maestro.
- No es mejor utilizar cuatro nodos maestros dedicados en lugar de tres y se pueden ocasionar problemas si se utilizan múltiples zonas de disponibilidad.
  - Si un nodo maestro produce un error, tiene quórum (3) para elegir un nuevo nodo. Si dos nodos producen un error, pierde ese quórum, de la misma forma que con tres nodos maestros dedicados.
  - En una configuración de tres zonas de disponibilidad, dos AZ tienen un nodo maestro dedicado y una AZ tiene dos. Si esa AZ experimenta alguna interrupción, las dos AZ restantes no tienen quórum necesario (3) para elegir un nuevo maestro.
- Contar con cinco nodos maestros dedicados funciona igual de bien que tres y permite mantener el quórum aunque pierda dos nodos. Sin embargo, dado que solo hay un nodo maestro dedicado activo en cada momento dado, esta configuración supone que usted paga por cuatro nodos inactivos. Muchos clientes consideran excesivo este nivel de protección de conmutación por error.

Si un clúster tiene un número par de nodos elegibles por el maestro, OpenSearch y Elasticsearch versiones 7.x y posteriores ignoran un nodo para que la configuración de votación sea siempre un número impar. En este caso, cuatro nodos maestros dedicados equivalen en esencia a tres (y dos, a uno).

> **Note**
>
> Si el clúster no tiene el quórum necesario para elegir un nuevo nodo maestro, se produce un error en las solicitudes de lectura y de escritura en el clúster. Este comportamiento difiere del predeterminado de OpenSearch.

Elección de tipos de instancias para nodos principales dedicados

Cuotas de dominio e instancia de OpenSearch Service

Si bien los nodos maestros dedicados no procesan solicitudes de búsqueda ni de consulta, su tamaño es proporcional al tamaño y el número de instancias, índices y particiones que son capaces de administrar. En el caso de los clústeres de producción, recomendamos como mínimo asignar los siguientes tipos de instancias para los nodos maestros dedicados.

Estas recomendaciones se basan en las cargas de trabajo habituales, pero pueden variar en función de sus necesidades. En los clústeres con muchas particiones o asignaciones de campo, puede resultar conveniente utilizar tipos de instancias más grandes. Para obtener más información, consulte Alarmas recomendadas de CloudWatch para Amazon OpenSearch Service para determinar si necesita usar un tipo de instancia más grande.

RAM	Compatibilidad máxima de nodos para Elasticsearch y OpenSearch Service de la versión 1.x a la 2.15	Compatibilidad máxima de partición para Elasticsearch y OpenSearch Service de la versión 1.x a la 2.15	Máximo de nodos admitidos para OpenSearch Service 2.17 y superiores	Máximo de particiones admitidas para OpenSearch Service 2.17 y posteriores
2 GB	No aplicable	No aplicable	10	1000
4 GB	No aplicable	No aplicable	10	5000
8 GB	10	10 000	30	15 000
16 GB	30	30 000	60	30 000
32 GB	75	40 000	120	60 000
64 GB	125	75 000	240	120 000
128 GB	200	75 000	480	240 000
256 GB	No aplicable	No aplicable	1002	500 000

Alarmas de CloudWatch recomendadas para Amazon OpenSearch Service

Las alarmas de CloudWatch realizan una acción cuando una métrica de CloudWatch supera un valor especificado para un periodo de tiempo determinado. Por ejemplo, es posible que desee que AWS envíe un email si el estado del clúster es `red` durante más de un minuto. En esta sección se incluyen algunas alarmas recomendadas para Amazon OpenSearch Service y cómo responder a ellas.

Puede implementar automáticamente estas alarmas mediante CloudFormation. Para ver una pila de ejemplo, consulte el repositorio de GitHub.

> **Note**
>
> Si implementa la pila de CloudFormation, las alarmas `KMSKeyError` y `KMSKeyInaccessible` existirán en un estado de `Insufficient Data`, ya que estas métricas solo aparecen si un dominio encuentra un problema con su clave de cifrado.

Para obtener más información sobre la configuración de las alarmas, consulte Creación de alarmas de Amazon CloudWatch en la Guía del usuario de Amazon CloudWatch.

Alarma	Problema
`ClusterStatus.red` El valor máximo de es >= 1 durante 1 minuto, 1 periodo consecutivo	Al menos una partición principal y sus réplicas no están asignados a un nodo. Consulte the section called “Estado rojo del clúster”.
El valor máximo de `ClusterStatus.yellow` es >= 1 durante 1 minuto, 5 periodos consecutivos	Al menos una partición de réplica no está asignada a un nodo. Consulte the section called “Estado amarillo del clúster”.
`FreeStorageSpace` El valor mínimo de es <=	El espacio de almacenamiento disponible de un nodo en su clúster se redujo hasta los 20 GiB. Consulte the section called “Falta de espacio de almacenamiento disponible”. Este valor está en MiB, por lo que,

Alarma	Problema
20 480 durante 1 minuto, 1 periodo consecutivo	en lugar de 20 480, recomendamos que lo configure en un 25 % del espacio de almacenamiento de cada nodo.
`ClusterIndexWritesBlocked` El valor de es >= 1 durante 5 minutos, 1 periodo consecutivo	El clúster bloquea las solicitudes de escritura. Consulte the section called “ClusterBlockException”.
El valor mínimo de `Nodes` es < x 1 día, 1 periodo consecutivo	x es el número de nodos del clúster. Esta alarma indica que al menos un nodo del clúster se mantuvo inaccesible durante un día. Consulte the section called “Nodos de clúster defectuosos”.

`AutomatedSnapshotFailure` El valor máximo de es >= 1 durante 1 minuto, 1 periodo consecutivo	Se produjo un error en una instantánea automatizada. Este error suele ser el resultado de un estado rojo del clúster. Consulte the section called “Estado rojo del clúster”.
	
	Para ver un resumen de todas las instantáneas automatizadas e información sobre los errores, también puede probar las siguientes solicitudes:
	
	`GET domain_endpoint /_snapshot/cs-automated/_all`
	`GET domain_endpoint /_snapshot/cs-automated-enc/_all`

El valor máximo de `CPUUtilization` o `WarmCPUUtilization` es >= 80 % durante 15 minutos, 3 periodos consecutivos	A veces puede producirse un uso de CPU del 100 %, pero el uso sostenido elevado es problemático. Considere la posibilidad de utilizar tipos de instancias más grandes o de agregar instancias.

Alarma	Problema
El valor máximo de `JVMMemory Pressure` es >= 95 % durante 1 minuto, 3 periodos consecutivos	El clúster podría encontrar errores de memoria insuficiente si aumenta el uso. Considere el escalado vertical. OpenSearch Service utiliza la mitad de la RAM de una instancia para la pila de Java, hasta un tamaño de 32 GiB. Puede escalar las instancias verticalmente hasta 64 GiB de RAM y después escalarlas horizontalmente mediante el agregado de instancias.
El valor máximo de `OldGenJVM MemoryPressure` es >= 80 % durante 1 minuto, 3 periodos consecutivos	
El valor máximo de `MasterCPU Utilization` es >= 50 % durante 15 minutos, 3 periodos consecutivos	Considere la posibilidad de utilizar tipos de instancias más grandes para los nodos maestros dedicados. Debido al rol que desempeñan en la estabilidad del clúster y las implementaciones azul/verde, los nodos maestros dedicados deberían tener un uso de CPU medio menor que los nodos de datos.
El valor máximo de `MasterJVM MemoryPressure` es >= 95 % durante 1 minuto, 3 periodos consecutivos	
El valor máximo de `MasterOld GenJVMMem oryPressure` es >= 80 % durante 1 minuto, 3 periodos consecutivos	

Alarma	Problema
`KMSKeyError` El valor de es >= 1 durante 1 minuto, 1 periodo consecutivo	La clave de cifrado de AWS KMS que se utiliza para cifrar los datos en reposo en el dominio está deshabilitada. Vuelva a habilitarla para restablecer las operaciones normales. Para más información, consulte the section called “Cifrado en reposo”.
`KMSKeyInaccessible` El valor de es >= 1 durante 1 minuto, 1 periodo consecutivo	La clave de cifrado de AWS KMS utilizada para cifrar los datos en reposo del dominio se eliminó o revocó sus permisos a OpenSearch Service. No puede recuperar los dominios que están en este estado. Sin embargo, si tiene una instantánea manual, puede utilizarla para migrar a un nuevo dominio. Para más información, consulte the section called “Cifrado en reposo”.
`shards.active` El valor de es >= 30 000 durante 1 minuto, 1 periodo consecutivo	El número total de particiones primarias y de réplicas activas es superior a 30 000. Es posible que esté rotando los índices con demasiada frecuencia. Considere la posibilidad de utilizar ISM para eliminar los índices una vez que alcancen una antigüedad determinada.
5xx alarmas >= 10 % de `OpenSearchRequests`	Es posible que uno o varios nodos de datos estén sobrecargados, o que las solicitudes no se completen dentro del periodo de tiempo de espera. Considere la posibilidad de cambiar a tipos de instancia más grandes o de agregar más nodos al clúster. Compruebe que está siguiendo las prácticas recomendadas para la arquitectura de particiones y clústeres.
El valor máximo de `MasterReachableFromNode` es < 1 durante 5 minutos, 1 periodo consecutivo	Esta alarma indica que el nodo maestro se ha detenido o es inaccesible. Estos errores suelen ser el resultado de un problema de conectividad de red o un problema de dependencia de AWS.
`ThreadpoolWriteQueue` El valor medio de es >= 100 durante 1 minuto, 1 periodo consecutivo	El clúster está experimentando una alta simultaneidad de indexación. Revise y controle las solicitudes de indexación, o aumente los recursos del clúster.

Alarma	Problema
`Threadpool SearchQueue` El valor medio de es >= 500 durante 1 minuto, 1 periodo consecutivo	El clúster está experimentando una alta simultaneidad de búsqueda. Considere la posibilidad de escalar el clúster. También se puede aumentar el tamaño de la cola de búsqueda, pero si se incrementa excesivamente puede provocar errores de memoria.
El valor máximo de `Threadpool SearchQueue` es >= 5000 durante 1 minuto, 1 periodo consecutivo	
El aumento de `Threadpool SearchRejected` SUM es >=1{expresión matemática DIFF ( )} durante 1 minuto, 1 periodo consecutivo	Estas alarmas le notifican los problemas del dominio que podrían afectar el rendimiento y la estabilidad.
El aumento de `Threadpool WriteRejected` SUM es >=1{expresión matemática DIFF ( )} durante 1 minuto, 1 periodo consecutivo	

> **Note**
>
> Si solo desea visualizar métricas, consulte the section called “Monitoreo de métricas del clúster”.

Otras alarmas para tener en cuenta

Considere la posibilidad de configurar las siguientes alarmas en función de las características de OpenSearch Service que utilice habitualmente.

Alarma	Problema
`WarmFreeStorageSpace` es >=10 %	Ha alcanzado el 10 % del total de almacenamiento intermedio gratuito. `WarmFreeStorageSpace` mide la suma del espacio de almacenamiento intermedio gratuito en MiB. UltraWarm utiliza Amazon S3 en lugar de discos adjuntos.
`HotToWarmMigrationQueueSize` El valor de es >= 20 durante 1 minuto, 3 periodos consecutivos	Un elevado número de índices se están trasladando simultáneamente del almacenamiento en caliente al UltraWarm. Considere la posibilidad de escalar el clúster.
`HotToWarmMigrationSuccessLatency` El valor de es >= 1 día, 1 periodo consecutivo	Si está intentando revertir los índices diarios, configure esta alarma para que se le notifique si el `HotToWarmMigrationSuccessCount` x latencia es superior a 24 horas.
El valor máximo de `WarmJVMMemoryPressure` es >= 95 % durante 1 minuto, 3 periodos consecutivos	El clúster podría encontrar errores de memoria insuficiente si aumenta el uso. Considere el escalado vertical. OpenSearch Service utiliza la mitad de la RAM de una instancia para la pila de Java, hasta un tamaño de 32 GiB. Puede escalar las instancias verticalmente hasta 64 GiB de RAM y después escalarlas horizontalmente mediante el agregado de instancias.
El valor máximo de `WarmOldGenJVMMemoryPressure` es >= 80 % durante 1	

Alarma	Problema
minuto, 3 periodos consecutivos	
`WarmToColdMigrationQueueSize` El valor de es >= 20 durante 1 minuto, 3 periodos consecutivos	Un elevado número de índices se están trasladando simultáneamente de UtraWarm a almacenamiento en frío. Considere la posibilidad de escalar el clúster.
`HotToWarmMigrationFailureCount` El valor de es >= 1 durante 1 minuto, 1 periodo consecutivo	Se pueden producir errores en las migraciones si se realizan durante instantáneas, reubicaciones de particiones o fusiones forzadas. Los errores durante las instantáneas o las reubicaciones de particiones suelen deberse a errores de nodo o problemas de conectividad de S3. La falta de espacio en el disco suele ser la causa subyacente de los errores en las fusiones forzosas.
`WarmToColdMigrationFailureCount` El valor de es >= 1 durante 1 minuto, 1 periodo consecutivo	Las migraciones suelen fallar cuando se produce un error en los intentos de migrar metadatos de índice al almacenamiento en frío. También pueden producirse errores al eliminar el estado de clúster de índice en caliente.
`WarmToColdMigrationLatency` El valor de es >= 1 día, 1 periodo consecutivo	Si está intentando revertir los índices diarios, configure esta alarma para que se le notifique si el `WarmToColdMigrationSuccessCount` x latencia es superior a 24 horas.
`AlertingDegraded` El valor de es >= 1 durante 1 minuto, 1 periodo consecutivo	El índice de alerta está en rojo, o uno o más nodos no ajustan a la programación.

Alarma	Problema
`ADPluginUnhealthy` El valor de es >= 1 durante 1 minuto, 1 periodo consecutivo	El complemento de detección de anomalías no funciona correctamente, ya sea debido a altas tasas de error o porque uno de los índices utilizados está en rojo.
`AsynchronousSearchFailureRate` El valor de es >= 1 durante 1 minuto, 1 periodo consecutivo	Al menos una búsqueda asíncrona ha fallado en el último minuto, lo que probablemente significa que el nodo coordinador ha fallado. El ciclo de vida de una solicitud de búsqueda asíncrona se administra únicamente en el nodo coordinador, por lo que si el coordinador cae, la solicitud falla.
`AsynchronousSearchStoreHealth` El valor de es >= 1 durante 1 minuto, 1 periodo consecutivo	El estado del almacén de respuestas de búsqueda asíncrona en el índice persistente está en rojo. Es posible que esté almacenando respuestas asíncronas de gran tamaño, lo que puede desestabilizar un clúster. Intente limitar las respuestas de búsqueda asíncronas a 10 MB o menos.
`SQLUnhealthy` El valor de es >= 1 durante 1 minuto, 3 periodos consecutivos	El complemento SQL está devolviendo códigos de respuesta 5xx o enviando un DSL de consulta no válido a OpenSearch. Solucione los problemas de las solicitudes que sus clientes hacen al complemento.
`LTRStatus.red` El valor de es >= 1 durante 1 minuto, 1 periodo consecutivo	Al menos uno de los índices necesarios para ejecutar el complemento Learning to Rank carece de particiones principales y no es funcional.

Referencia general para Amazon OpenSearch Service

Amazon OpenSearch Service es compatible con una gran variedad de instancias, operaciones, complementos y otros recursos.

Temas

- Tipos de instancia compatibles con Amazon OpenSearch Service
- Características según la versión de motor en Amazon OpenSearch Service
- Complementos según la versión de motor en Amazon OpenSearch Service
- Operaciones compatibles en Amazon OpenSearch Service
- Amazon OpenSearch Service quotas
- Instancias reservadas en Amazon OpenSearch Service
- Otros recursos admitidos en Amazon OpenSearch Service

Tipos de instancia compatibles con Amazon OpenSearch Service

Amazon OpenSearch Service es compatible con los siguientes tipos de instancias. No todas las regiones admiten todos los tipos de instancias. Para obtener más información sobre la disponibilidad, consulte Precio de Amazon OpenSearch Service.

Para obtener información sobre qué tipo de instancia es adecuada para un caso de uso, consulte the section called “Determinación del tamaño de dominios”, the section called “Cuotas de tamaño del volumen de EBS” y the section called “Cuotas de red”.

Tipos de instancias de generación actual

Para lograr un mejor rendimiento, le recomendamos que use los siguientes tipos de instancias al crear nuevos dominios de OpenSearch Service.

Tipo de instancia	instancias	Restricciones
i4i	i4i.large .search	Los tipos de instancias i4i necesitan Elasticsearch 5.1 o una versión posterior o cualquier versión de OpenSearch y no son compatibles con el almacenamiento en volúmenes de EBS.

Tipo de instancia	instancias	Restricciones
	i4i.xlarge.search i4i.2xlarge.search i4i.4xlarge.search i4i.8xlarge.search i4i.12xlarge.search i4i.16xlarge.search i4i.24xlarge.search i4i.32xlarge.search	

Tipo de instancia	instancias	Restricciones
i4g	i4g.large.search i4g.xlarge.search i4g.2xlarge.search i4g.4xlarge.search i4g.8xlarge.search i4g.16xlarge.search	Los tipos de instancias i4g necesitan Elasticsearch 7.9 o una versión posterior o cualquier versión de OpenSearch y no son compatibles con los volúmenes de almacenamiento de EBS.

Tipo de instancia	instancias	Restricciones
i8g	i8g.large.search i8g.xlarge.search i8g.2xlarge.search i8g.4xlarge.search i8g.8xlarge.search i8g.12xlarge.search i8g.16xlarge.search	Los tipos de instancias i8g necesitan Elasticsearch 7.9 o una versión posterior o cualquier versión de OpenSearch y no son compatibles con los volúmenes de almacenamiento de EBS.

Tipo de instancia	instancias	Restricciones
Graviton3	C7g.large.search C7g.xlarge.search C7g.2xlarge.search C7g.4xlarge.search C7g.8xlarge.search C7g.12xlarge.search C7g.16xlarge.search M7g.large.search M7g.xlarge.search M7g.2xlarge.search M7g.4xlarge.search M7g.8xlarge.search	Graviton3 solo es compatible con GP3.

Tipo de instancia	instancias	Restricciones
	M7g.12xlarge.search M7g.16xlarge.search R7g.medium.search R7g.large.search R7g.xlarge.search R7g.2xlarge.search R7g.4xlarge.search R7g.8xlarge.search R7g.12xlarge.search R7g.16xlarge.search R7gd.large.search R7gd.xlarge.search R7gd.2xlarge.search	

Tipo de instancia	instancias	Restricciones
	R7gd.4xlarge.search R7gd.8xlarge.search R7gd.12xlarge.search R7gd.16xlarge.search	

Tipo de instancia	instancias	Restricciones
OR1	or1.medium.search or1.large.search or1.xlarge.search or1.2xlarge.search or1.4xlarge.search or1.8xlarge.search or1.12xlarge.search or1.16xlarge.search	• Los tipos de instancia OR1 requieren OpenSearch 2.11 o una versión posterior. • Las instancias OR1 solo son compatibles con otros tipos de instancias Graviton de nodos maestros (C6g, M6g, R6g).

Tipo de instancia	instancias	Restricciones
OR2	or2.medium.search or2.large.search or2.xlarge.search or2.2xlarge.search or2.4xlarge.search or2.8xlarge.search or2.12xlarge.search or2.16xlarge.search	

Tipo de instancia	instancias	Restricciones

OM2	`om2.large.search`
	`om2.xlarge.search`
	`om2.2xlarge.search`
	`om2.4xlarge.search`
	`om2.8xlarge.search`
	`om2.12xlarge.search`
	`om2.16xlarge.search`	

Tipo de instancia	instancias	Restricciones

Im4gn	`im4gn.large.search`
	`im4gn.xlarge.search`
	`im4gn.2xlarge.search`
	`im4gn.4xlarge.search`
	`im4gn.8xlarge.search`
	`im4gn.16xlarge.search`
		• Los tipos de instancias Im4gn necesitan Elasticsearch 7.9 o posterior o cualquier versión de OpenSearch y no son compatibles con los volúmenes de almacenamiento de EBS.
		• Las instancias Im4gn solo son compatibles con otros tipos de instancias Graviton (C6g, M6g, R6g, R6gd). No se pueden combinar instancias Graviton y no Graviton en el mismo clúster.

Tipo de instancia	instancias	Restricciones

C5	`c5.large.search`
	`c5.xlarge.search`
	`c5.2xlarge.search`
	`c5.4xlarge.search`
	`c5.9xlarge.search`
	`c5.18xlarge.search`
		Los tipos de instancias C5 requieren Elasticsearch 5.1 o posterior o cualquier versión de OpenSearch.

Tipo de instancia	instancias	Restricciones

C6g	`c6g.large.search`
	`c6g.xlarge.search`
	`c6g.2xlarge.search`
	`c6g.4xlarge.search`
	`c6g.8xlarge.search`
	`c6g.12xlarge.search`
		• Los tipos de instancia C6g requieren Elasticsearch 7.9 o posterior o cualquier versión de OpenSearch.
		• Las instancias C6g solo son compatibles con otros tipos de instancias Graviton (Im4gn, M6g, R6g, R6gd). No se pueden combinar instancias Graviton y no Graviton en el mismo clúster.

Tipo de instancia	instancias	Restricciones

I3	`i3.large.search`
	`i3.xlarge.search`
	`i3.2xlarge.search`
	`i3.4xlarge.search`
	`i3.8xlarge.search`
	`i3.16xlarge.search`	

M5	`m5.large.search`
	`m5.xlarge.search`
	`m5.2xlarge.search`
	`m5.4xlarge.search`
	`m5.12xlarge.search`
		Los tipos de instancias M5 requieren Elasticsearch 5.1 o posterior o cualquier versión de OpenSearch.

Tipo de instancia	instancias	Restricciones

M6g	`m6g.large.search`
	`m6g.xlarge.search`
	`m6g.2xlarge.search`
	`m6g.4xlarge.search`
	`m6g.8xlarge.search`
	`m6g.12xlarge.search`
		• Los tipos de instancia M6g requieren Elasticsearch 7.9 o posterior o cualquier versión de OpenSearch.
		• Las instancias M6g solo son compatibles con otros tipos de instancias Graviton (Im4gn, C6g, R6g, R6gd). No se pueden combinar instancias Graviton y no Graviton en el mismo clúster.

Tipo de instancia	instancias	Restricciones

R5	`r5.large.search`
	`r5.xlarge.search`
	`r5.2xlarge.search`
	`r5.4xlarge.search`
	`r5.12xlarge.search`
		Los tipos de instancias R5 requieren Elasticsearch 5.1 o posterior o cualquier versión de OpenSearch.

Tipo de instancia	instancias	Restricciones

R6g	`r6g.large.search`
	`r6g.xlarge.search`
	`r6g.2xlarge.search`
	`r6g.4xlarge.search`
	`r6g.8xlarge.search`
	`r6g.12xlarge.search`
		• Los tipos de instancia R6g requieren Elasticsearch 7.9 o posterior o cualquier versión de OpenSearch.
		• Las instancias R6g solo son compatibles con otros tipos de instancias Graviton (Im4gn, C6g, M6g, R6gd). No se pueden combinar instancias Graviton y no Graviton en el mismo clúster.

Tipo de instancia	instancias	Restricciones

R6gd	`r6gd.large.search`
	`r6gd.xlarge.search`
	`r6gd.2xlarge.search`
	`r6gd.4xlarge.search`
	`r6gd.8xlarge.search`
	`r6gd.12xlarge.search`
	`r6gd.16xlarge.search`
		• Los tipos de instancias R6gd necesitan Elasticsearch 7.9 o posterior o cualquier versión de OpenSearch y no son compatibles con los volúmenes de almacenamiento de EBS.
		• Las instancias R6gd solo son compatibles con otros tipos de instancias Graviton (Im4gn, C6g, M6g, R6g). No se pueden combinar instancias Graviton y no Graviton en el mismo clúster.

Tipo de instancia	instancias	Restricciones

T3	`t3.small.search`
	
	`t3.medium.search`
		• Los tipos de instancia T3 requieren Elasticsearch 5.6 o posterior o cualquier versión de OpenSearch.
		• Puede usar los tipos de instancia T3 solo si el dominio está aprovisionado sin modo de espera. Para obtener más información, consulte the section called “Multi-AZ sin modo de espera”.
		• Solo puede usar los tipos de instancias T3 cuando el recuento de instancias del dominio sea 10 o menos.
		• Los tipos de instancias T3 no son compatibles con el almacenamiento UltraWarm, el almacenamiento en frío o el ajuste automático.

Tipo de instancia	instancias	Restricciones

c7i	`c7i.large.search`
	`c7i.xlarge.search`
	`c7i.2xlarge.search`
	`c7i.4xlarge.search`
	`c7i.8xlarge.search`
	`c7i.12xlarge.search`
	`c7i.16xlarge.search`
		• La instancia c7i necesita Elasticsearch 5.1 o posterior o cualquier versión de OpenSearch y solo es compatible con los volúmenes de almacenamiento de GP3.

Tipo de instancia	instancias	Restricciones

m7i	`m7i.large.search`
	`m7i.xlarge.search`
	`m7i.2xlarge.search`
	`m7i.4xlarge.search`
	`m7i.8xlarge.search`
	`m7i.12xlarge.search`
	`m7i.16xlarge.search`
		• La instancia m7i necesita Elasticsearch 5.1 o posterior o cualquier versión de OpenSearch y solo es compatible con los volúmenes de almacenamiento de GP3.

Tipo de instancia	instancias	Restricciones

r7i	`r7i.large.search`
	`r7i.xlarge.search`
	`r7i.2xlarge.search`
	`r7i.4xlarge.search`
	`r7i.8xlarge.search`
	`r7i.12xlarge.search`
	`r7i.16xlarge.search`
		• La instancia r7i necesita Elasticsearch 5.1 o posterior o cualquier versión de OpenSearch y solo es compatible con los volúmenes de almacenamiento de GP3.

Tipo de instancia	instancias	Restricciones

c8g	`c8g.large`
	`c8g.xlarge`
	`c8g.2xlarge`
	`c8g.4xlarge`
	`c8g.8xlarge`
	`c8g.12xlarge`
	`c8g.16xlarge`
		• Los tipos de instancias c8g necesitan Elasticsearch 7.9 o posterior o cualquier versión de OpenSearch y solo son compatibles con los volúmenes de almacenamiento de GP3.

Tipo de instancia	instancias	Restricciones

m8g	`m8g.medium`
	`m8g.large`
	`m8g.xlarge`
	`m8g.2xlarge`
	`m8g.4xlarge`
	`m8g.8xlarge`
	`m8g.12xlarge`
	`m8g.16xlarge`
		• Los tipos de instancias m8g necesitan Elasticsearch 7.9 o posterior o cualquier versión de OpenSearch y solo son compatibles con los volúmenes de almacenamiento de GP3.

Tipo de instancia	instancias	Restricciones

r8g	`r8g.medium`
	`r8g.large`
	`r8g.xlarge`
	`r8g.2xlarge`
	`r8g.4xlarge`
	`r8g.8xlarge`
	`r8g.12xlarge`
	`r8g.16xlarge`
		• Los tipos de instancias r8g necesitan Elasticsearch 7.9 o posterior o cualquier versión de OpenSearch y solo son compatibles con los volúmenes de almacenamiento de GP3.

Tipo de instancia	instancias	Restricciones

r8gd	`r8gd.medium`
	`r8gd.large`
	`c8g.xlarge`
	`r8gd.2xlarge`
	`r8gd.4xlarge`
	`r8gd.8xlarge`
	`r8gd.12xlarge`
	`r8gd.16xlarge`
		• Los tipos de instancias r8gd necesitan Elasticsearch 7.9 o posterior o cualquier versión de OpenSearch y no son compatibles con los volúmenes de almacenamiento de EBS.

Tipos de instancias de generación anterior

OpenSearch Service ofrece tipos de instancia de generaciones anteriores para los usuarios que hayan optimizado sus aplicaciones con estas instancias y aún no hayan actualizado. Le recomendamos que utilice los tipos de instancia de la generación actual para obtener un rendimiento óptimo, pero seguimos admitiendo los siguientes tipos de instancia de la generación anterior.

Tipo de instancia	instancias	Restricciones

C4	`c4.large.search`
	`c4.xlarge.search`
	`c4.2xlarge.search`
	`c4.4xlarge.search`
	`c4.8xlarge.search`	

I2	`i2.xlarge.search` `i2.2xlarge.search`	

M3	`m3.medium.search`
	`m3.large.search`
	`m3.xlarge.search`
	`m3.2xlarge.search`
		• Los tipos de instancia M3 no admiten el cifrado de datos en reposo, el control de acceso detallado ni la búsqueda en clústeres.
		• Los tipos de instancia M3 tienen restricciones adicionales por la versión OpenSearch. Para más información, consulte the section called “Tipo de instancia M3 no válido”.

M4	`m4.large.search`	

Tipo de instancia	instancias	Restricciones

	`m4.xlarge.search`
	`m4.2xlarge.search`
	`m4.4xlarge.search`
	`m4.10xlarge.search`	

R3	`r3.large.search`
	`r3.xlarge.search`
	`r3.2xlarge.search`
	`r3.4xlarge.search`
	`r3.8xlarge.search`
		Los tipos de instancia R3 no admiten el cifrado de datos en reposo ni el control de acceso detallado.

Tipo de instancia	instancias	Restricciones

R4	`r4.large.search`
	`r4.xlarge.search`
	`r4.2xlarge.search`
	`r4.4xlarge.search`
	`r4.8xlarge.search`
	`r4.16xlarge.search`	

T2	`t2.micro.search`
	`t2.small.search`
	`t2.medium.search`
		• Solo puede usar los tipos de instancias T2 cuando el recuento de instancias del dominio sea 10 o menos.
		• El tipo de instancia `t2.micro.search` solo admite Elasticsearch 1.5 y 2.3.
		• Los tipos de instancia T2 no son compatibles con el cifrado de datos en reposo, el control de acceso detallado, el almacenamiento UltraWarm, el almacenamiento en frío, la búsqueda entre clústeres o el ajuste automático.

Tip

Recomendamos utilizar tipos de instancias diferentes para nodos maestros dedicados y nodos de datos.

Características según la versión de motor en Amazon OpenSearch Service

Muchas características de OpenSearch Service tienen un requisito mínimo de versión de OpenSearch o un requisito de versión heredado de Elasticsearch OSS. Si cumple la versión mínima de una característica, pero la característica no está disponible en el dominio, actualice el software de servicio.

Característica	Versión mínima requerida de OpenSearch	Versión mínima requerida de Elasticsearch
Origen derivado	3.1	No incluido
Índice Star-tree		
Compatibilidad con Amazon Q	2.17	No incluido
Complementos personalizados	2.15	No incluido
Búsqueda de segmentos simultánea	2.13	No incluido
Generación de consultas en lenguaje natural	2.13	No incluido

Característica	Versión mínima requerida de OpenSearch	Versión mínima requerida de Elasticsearch
Búsqueda semántica multimodal	2.11	No incluido
Orígenes de datos de consulta directa	2.11	No incluido
Nuevos conectores de machine learning	2.9	No incluido
Búsqueda semántica	2.9	No incluido
Búsqueda puntual	2,5	No incluido
Notificaciones	2.3	No incluido
ML Commons	1.3	No incluido
Replicación entre clústeres	1.1	7.10
Transformaciones de índices	1.0	No incluido

Característica	Versión mínima requerida de OpenSearch	Versión mínima requerida de Elasticsearch
Nodo coordinador dedicado	1.0	6.8
Compatibilidad con VPC	1.0	1.0
Requerir HTTPS para todo el tráfico al dominio		
Compatibilidad con Multi-AZ		
Nodos maestros dedicados		
Paquetes personalizados		
Puntos de conexión personalizados		
Publicación de registros lentos		

Característica	Versión mínima requerida de OpenSearch	Versión mínima requerida de Elasticsearch
Registro de errores de publicación	1.0	5.1
Cifrado de datos en reposo		
Autenticación Cognito para OpenSearch Dashboards		
Actualizaciones locales		
Instantáneas automatizadas por hora	1.0	5.3
Cifrado de nodo a nodo	1.0	6.0
Compatibilidad con cliente REST de alto nivel de Java		

Característica	Versión mínima requerida de OpenSearch	Versión mínima requerida de Elasticsearch
Compresión de solicitud y respuesta HTTP		
Alertas	1.0	6.2
SQL	1.0	6.5
Búsqueda en clústeres	1.0	6.7
Control de acceso detallado		
Autenticación SAML para OpenSearch Dashboards		
Ajuste automático		
Reindexación remota		
UltraWarm	1.0	6.8
Administración de estados de índice		

Característica	Versión mínima requerida de OpenSearch	Versión mínima requerida de Elasticsearch
k-NN por distancia euclidiana	1.0	7.1
Detección de anomalías	1.0	7.4
k-NN por similitud coseno	1.0	7.7
Aprender a clasificar		
Lenguaje de procesamiento de canalizaciones	1.0	7.9
Informes de OpenSearch Dashboards		
Análisis de seguimiento de OpenSearch Dashboards		

Característica	Versión mínima requerida de OpenSearch	Versión mínima requerida de Elasticsearch
Instancias de Graviton basadas en ARM		
Almacenamiento en frío		
Distancia de Hamming, distancia de norma L1 y scripting Painless para k-NN	1.0	7.10
Búsqueda asíncrona		
Compatibilidad con Curator	No incluido	5.1

Para obtener información acerca de los complementos, que habilitan algunas de estas características y funcionalidades adicionales, consulte the section called “Complementos por versión de motor”. Para obtener información sobre la API de OpenSearch para cada versión, consulte the section called “Operaciones admitidas”.

Complementos según la versión de motor en Amazon OpenSearch Service

Los dominios de Amazon OpenSearch Service vienen empaquetados con complementos de la comunidad de OpenSearch. El servicio implementa y administra automáticamente los complementos,

pero los que implementa son diferentes en función de la versión de OpenSearch o Elasticsearch OSS heredado que haya elegido para el dominio.

En la siguiente tabla se enumeran los complementos por versión de OpenSearch, así como las versiones compatibles de Elasticsearch OSS heredado. Solo incluye complementos con los que puede interactuar; no es exhaustivo. OpenSearch Service utiliza complementos adicionales a fin de habilitar la funcionalidad del servicio principal, como el complemento S3 Repository para instantáneas y el complemento OpenSearch Performance Analyzer con fines de optimización y supervisión. Para obtener una lista completa de todos los complementos que se ejecutan en un dominio, realice la siguiente solicitud:

```
GET _cat/plugins?v
```

Complemento	Versión mínima requerida de OpenSearch	Versión mínima requerida de Elasticsearch
HanLP	2.11	No admitido
Análisis de hebreo	2.11	No admitido
Search Ranking de Amazon Personalize	2.9	No admitido
Búsqueda neuronal	2.9	No admitido
Security Analytics	2,5	No admitido
Notificaciones de OpenSearch	2.3	No admitido

Complemento	Versión mínima requerida de OpenSearch	Versión mínima requerida de Elasticsearch
ML Commons	1.3	No admitido
Análisis de Sudachi (recomendado para japonés)	1.3	No admitido
STConvert	1.3	No admitido
Análisis de Pinyin	1.3	No admitido
Análisis de Nori	1.3	No admitido
Observabilidad de OpenSearch	1.2	No admitido
Replicación entre clústeres de OpenSearch	1.1	7.10
Búsqueda asíncrona de OpenSearch	1.0	7.10
IK (Chinese) Analysis	1.0	7.7

Complemento	Versión mínima requerida de OpenSearch	Versión mínima requerida de Elasticsearch
Vietnamese Analysis		
Thai Analysis		
Learning to Rank		
Detección de anomalías de OpenSearch	1.0	7.4
k-NN de OpenSearch	1.0	7.1
Administración de estados de índices de OpenSearch	1.0	6.8
Seguridad de OpenSearch	1.0	6.7
SQL de OpenSearch	1.0	6.5

Complemento	Versión mínima requerida de OpenSearch	Versión mínima requerida de Elasticsearch
Alertas de OpenSearch	1.0	6.2
Ukrainian Analysis	1.0	5.3
Mapper Size	1.0	5.3
Mapper Murmur3	1.0	5.1
Ingest User Agent Processor	1.0	5.1
Ingest Attachment Processor	1.0	5.1
Stempel Polish Analysis	1.0	5.1
Smart Chinese Analysis	1.0	5.1
Seunjeon Korean Analysis	1.0	5.1
Phonetic Analysis	1.0	2.3

Complemento	Versión mínima requerida de OpenSearch	Versión mínima requerida de Elasticsearch
Japanese (kuromoji) Analysis	1.0	Incluido en todos los dominios
ICU Analysis	1.0	Incluido en todos los dominios

Complementos opcionales

Además de los complementos predeterminados que vienen preinstalados, Amazon OpenSearch Service admite varios complementos de análisis de lenguaje opcionales. Puede usar Consola de administración de AWS y AWS CLI para asociar un complemento a un dominio, desasociar un complemento de un dominio y enumerar todos los complementos. Un paquete de complementos opcionales es compatible con una versión específica de OpenSearch y solo se puede asociar a los dominios con esa versión.

Tenga en cuenta que para el complemento Sudachi, al volver a asociar un archivo de diccionario, no se refleja inmediatamente en el dominio. El diccionario se actualiza cuando se ejecuta la siguiente implementación azul/verde en el dominio como parte de un cambio de configuración u otra actualización. Como alternativa, puede crear un nuevo paquete con los datos actualizados, crear un nuevo índice con un nuevo paquete, volver a indexar el índice existente en el nuevo índice y, a continuación, eliminar el índice anterior. Si prefiere utilizar el enfoque de reindexación, utilice un alias de índice para que no se interrumpa el tráfico.

Los complementos opcionales utilizan el tipo de paquete `ZIP-PLUGIN`. Para obtener más información sobre los complementos opcionales, consulte the section called “Paquetes”.

Operaciones compatibles en Amazon OpenSearch Service

OpenSearch Service admite muchas versiones de OpenSearch y Elasticsearch OSS heredado. Las siguientes secciones muestran las operaciones compatibles con cada versión de OpenSearch Service.

Temas

• Diferencias de API destacadas

Diferencias de API destacadas

Nuevas API de lista

Para admitir clústeres grandes con un gran número de índices y particiones, hemos introducido nuevas API de lista compatibles con la paginación, como _list/indices y _list/shards. La API List recupera las estadísticas sobre los índices y los particiones en un formato paginado. Esto agiliza la tarea de procesar las respuestas que incluyen muchos índices.

- `_list/indices`: _list/indices
- `_list/shards`: _list/shards

Cambios en las API existentes

Para admitir clústeres de gran tamaño, hemos incorporado a la API `_cluster/stats` la posibilidad de agregar filtros de métricas adicionales que permitan recuperar solo las respuestas estadísticas relevantes, como `_cluster/stats/<metric>/nodes/<node-filters>` y `_cluster/stats/<metric>/<index_metric>/nodes/<node-filters>`. Para obtener más información, consulte _cluster/stats.

Hemos agregado soporte en la API `_cat/shards` para la cancelación de tareas con la especificación de un parámetro de solicitud `cancel_after_time_interval`. Para obtener más información, consulte _cat/shards.

Limitación del tamaño de la respuesta para la API _cat

Para admitir clústeres grandes con un número total de instancias superior a 200 en nodos de datos y calientes, tenemos un límite de 10 000 en la cantidad de índices devueltos por la `_cat/segments API`. Si el número de índices de la respuesta supera este límite, la API devuelve un error 429. Para evitarlo, puede especificar un filtro de patrón de índice en la consulta, por ejemplo `_cat/segments/<index-pattern>`.

Configuración y estadísticas

OpenSearch Service solo es compatible con solicitudes PUT a la API `_cluster/settings` que utilizan el formato de configuración “plano”. Rechaza las solicitudes que utilizan el formato de configuración expandido.

```
// Accepted
PUT _cluster/settings
{
  "persistent" : {
    "action.auto_create_index" : false
  }
}

// Rejected
PUT _cluster/settings
{
  "persistent": {
    "action": {
      "auto_create_index": false
    }
  }
}
```

El cliente REST Java de alto nivel utiliza el formato expandido, por lo que, si necesita enviar solicitudes de configuración, use el cliente de bajo nivel.

Antes de Elasticsearch 5.3, la API `_cluster/settings` en dominios de OpenSearch Service solo era compatible con el método HTTP `PUT` y no con el método `GET`. OpenSearch y las versiones posteriores de Elasticsearch son compatibles con el método `GET`, tal y como se muestra en el ejemplo siguiente:

```
GET https://domain-name.region.es.amazonaws.com/_cluster/settings?pretty
```

A continuación se muestra un ejemplo de rentabilidad:

```
{
  "persistent": {
    "cluster": {
      "routing": {
        "allocation": {
          "cluster_concurrent_rebalance": "2",
          "node_concurrent_recoveries": "2",
          "disk": {
            "watermark": {
              "low": "1.35gb",
              "flood_stage": "0.45gb",
              "high": "0.9gb"
```

```
            }
          },
          "node_initial_primarirecoveries": "4"
        }
      }
    },
    "indices": {
      "recovery": {
        "max_bytper_sec": "40mb"
      }
    }
  }
}
```

Si compara las respuestas de un clúster de OpenSearch de código abierto y OpenSearch Service para determinadas API de configuración y estadísticas, es posible que observe que faltan campos. OpenSearch Service redacta cierta información que expone los elementos internos del servicio, como la ruta de acceso de datos del sistema de archivos desde `_nodes/stats` o el nombre del sistema operativo y la versión de `_nodes`.

Reducir

La API `_shrink` puede hacer que las actualizaciones, los cambios de configuración y las eliminaciones de dominio no se lleven a cabo correctamente. No recomendamos utilizarla en dominios que ejecuten las versiones 5.3 o 5.1 de Elasticsearch. Estas versiones tienen un error que puede causar un error en la restauración de instantáneas de índices encogidos.

Si utiliza la API `_shrink` en otras versiones de Elasticsearch o de OpenSearch, realice la siguiente solicitud antes de comenzar la operación de reducción:

```
PUT https://domain-name.region.es.amazonaws.com/source-index/_settings
{
  "settings": {
    "index.routing.allocation.require._name": "name-of-the-node-to-shrink-to",
    "index.blocks.read_only": true
  }
}
```

A continuación, realice la siguiente solicitud después de completar la operación de reducción:

```
PUT https://domain-name.region.es.amazonaws.com/source-index/_settings
```

```
{
  "settings": {
    "index.routing.allocation.require._name": null,
    "index.blocks.read_only": false
  }
}

PUT https://domain-name.region.es.amazonaws.com/shrunken-index/_settings
{
  "settings": {
    "index.routing.allocation.require._name": null,
    "index.blocks.read_only": false
  }
}
```

Nuevas API de lista

Para admitir clústeres grandes con una gran cantidad de índices y particiones, hemos introducido nuevas API de lista compatibles con la paginación, es decir, `_list/indices` y `_list/shards`. La API List recupera las estadísticas sobre los índices y los particiones en un formato paginado. Esto agiliza la tarea de procesar las respuestas que incluyen muchos índices. Para obtener más información acerca de `_list/indices`, consulte List indices. Para obtener más información acerca de `_list/shards`, consulte List shards.

Cambios en las API existentes

Para admitir clústeres grandes, hemos agregado soporte en `_cluster/stats/<metric>/nodes/<node-filters>` y`_cluster/stats/<metric>/<index_metric>/nodes/<node-filters>`. Para obtener más información sobre `_cluster/stats`, consulte Estadísticas de clústeres.

Limitar el tamaño de respuesta de las API _cat

Para admitir clústeres grandes con un número total de instancias superior a 200 en nodos de datos y calientes, tenemos un límite de 10 000 en la cantidad de índices que devuelve la API _cat/segments. Si el número de índices de la respuesta supera este límite, la API devuelve un error 429. Para evitarlo, puede especificar un filtro de patrón de índice en la consulta (por ejemplo, `_cat/segments/<index-pattern>`).

Además, la API `_cat/shards` ahora admite la cancelación de tareas especificando el parámetro de solicitud `cancel_after_time_interval`. Para obtener más información acerca de las particiones, consulte CAT shards.

Elección de los tipos de instancias para nodos maestro dedicados

En la siguiente tabla, se proporcionan recomendaciones de elección según los tipos de instancia adecuados para los nodos maestros dedicados:

RAM	Número máximo de nodos compatibles	Cantidad máxima de particiones admitidas	
2 GB	10	1 000	
4 GB	10	5 000	
8 GB	30	15.000	
16 GB	60	30.000	
32 GB	120	60 000	
64 GB	240	120 000	
128 GB	480	240 000	
256 GB	1002	500.000	

OpenSearch versión 2.19

Para obtener información sobre las operaciones de OpenSearch 2.19, consulte la referencia de la API de REST de OpenSearch o la referencia de la API del complemento específico. Para obtener más información acerca de los cambios de esta versión, consulte las notas de la versión 2.19.

OpenSearch versión 2.17

En OpenSearch 2.17, OpenSearch Service admite las siguientes operaciones. Para obtener información sobre la mayoría de las operaciones, consulte OpenSearch REST API reference, o la referencia de la API del complemento específico.

> **Note**
>
> El ajuste de `cluster.max_shards_per_node` no se puede modificar a partir de OpenSearch 2.17. Para OpenSearch 2.17 y versiones posteriores, OpenSearch Service admite 1000 particiones por cada 16 GB de memoria de heap de la JVM, hasta un máximo de 4000 particiones por nodo.

- Todas las operaciones de la ruta de índice (como */index-name* `/_forcemerge` , */index-name* `/update/`*id* y */index-name* `/_close`)
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (excepto `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/settings` para varias propiedades[4]:
  - `action.auto_create_index`
  - `action.search.shard_count.limit`
  - `indices.breaker.fielddata.limit`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_list`
- `/_ltr`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_plugins/_asynchronous_search`
- `/_plugins/_alerting`
- `/_plugins/_anomaly_detection`
- `/_plugins/_ism`
- `/_plugins/_ml`
- `/_plugins/_notifications`
- `/_plugins/_ppl`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_resolve/index`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search/pipeline`
- `/_search/point_in_time`
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`
- `/_split`
- `/_stats`
- `/_status`
- `/_tasks`
- `/_template`
- `/_update_by_query` [1]
- `/_validate`

  - indices.breaker.request.limit
  - indices.breaker.total.limit
  - cluster.search.request.slowlog.level
  - cluster.search.request.slowlog.threshold.warn
  - cluster.search.request.slowlog.threshold.info
  - cluster.search.request.slowlog.threshold.debug
  - cluster.search.request.slowlog.threshold.trace
  - search.phase_took_enabled
- /_cluster/state
- /_cluster/stats
- /_count
- /_dashboards
- /_plugins/_security
- /_plugins/_security_analytics
- /_plugins/_sm
- /_plugins/_sql
- /_percolate
- /_rank_eval

1. Los cambios en la configuración de un clúster pueden interrumpir estas operaciones antes de que finalicen. Recomendamos utilizar la operación `/_tasks` combinada con estas operaciones para verificar que las solicitudes se hayan llevado a cabo correctamente.
2. Las solicitudes DELETE para `/_search/scroll` con un cuerpo de mensaje deben especificar `"Content-Length"` en el encabezado HTTP. La mayoría de los clientes agregan este encabezado de forma predeterminada. Para evitar problemas con caracteres = en valores `scroll_id`, utilice el cuerpo de la solicitud, no la cadena de solicitud, para pasar valores de `scroll_id` a OpenSearch Service.

3. Consulte the section called “Otros recursos admitidos” para ver consideraciones sobre la utilización de scripts.
4. Se refiere al método PUT. Para obtener información acerca del método GET, consulte the section called “Diferencias de API destacadas”. Esta lista solo hace referencia a las operaciones genéricas de OpenSearch compatibles con OpenSearch Service y no incluye operaciones compatibles con complementos específicos para detección de anomalías, ISM, etc.
5. Consulte the section called “Reducir”.

> **Note**
>
> Actualmente, el cambio de la funcionalidad del ajuste `cluster.max_shards_per_node` no está habilitado para los clientes con Multi-AZ (zona de disponibilidad) en espera.

OpenSearch versión 2.15

En OpenSearch 2.15, OpenSearch Service admite las siguientes operaciones. Para obtener información sobre la mayoría de las operaciones, consulte OpenSearch REST API reference, o la referencia de la API del complemento específico.

- Todas las operaciones de la ruta de índice (como */index-name* `/_forcemerge` , */index-name* `/update/`*id* y */index-name* `/_close`)
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (excepto `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_ltr`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_plugins/_asynchronous_search`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_resolve/index`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search/pipeline`
- `/_search/point_in_time`
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`

- /_cluster/pending_tasks
- /_cluster/settings para varias propiedades[4]:
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
  - cluster.max_shards_per_node
  - cluster.search.request.slowlog.level
  - cluster.search.request.slowlog.threshold.warn
  - cluster.search.request.slowlog.threshold.info
  - cluster.search.request.slowlog.threshold.debug
  - cluster.search.request.slowlog.threshold.trace
  - search.phase_took_enabled
- /_cluster/state
- /_cluster/stats
- /_plugins/_alerting
- /_plugins/_anomaly_detection
- /_plugins/_ism
- /_plugins/_ml
- /_plugins/_notifications
- /_plugins/_ppl
- /_plugins/_security
- /_plugins/_security_analytics
- /_plugins/_sm
- /_plugins/_sql
- /_percolate
- /_rank_eval
- /_split
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

- `/_count`
- `/_dashboards`

1. Los cambios en la configuración de un clúster pueden interrumpir estas operaciones antes de que finalicen. Recomendamos utilizar la operación `/_tasks` combinada con estas operaciones para verificar que las solicitudes se hayan llevado a cabo correctamente.
2. Las solicitudes DELETE para `/_search/scroll` con un cuerpo de mensaje deben especificar `"Content-Length"` en el encabezado HTTP. La mayoría de los clientes agregan este encabezado de forma predeterminada. Para evitar problemas con caracteres = en valores `scroll_id`, utilice el cuerpo de la solicitud, no la cadena de solicitud, para pasar valores de `scroll_id` a OpenSearch Service.
3. Consulte para ver consideraciones sobre la utilización de scripts the section called “Otros recursos admitidos”.
4. Se refiere al método `PUT`. Para obtener información acerca del método `GET`, consulte the section called “Diferencias de API destacadas”. Esta lista solo hace referencia a las operaciones genéricas de OpenSearch compatibles con OpenSearch Service y no incluye operaciones compatibles con complementos específicos para detección de anomalías, ISM, etc.
5. Consulte the section called “Reducir”.

OpenSearch versión 2.13

En OpenSearch 2.13, OpenSearch Service admite las siguientes operaciones. Para obtener información sobre la mayoría de las operaciones, consulte OpenSearch REST API reference, o la referencia de la API del complemento específico.

- Todas las operaciones de la ruta de índice (como */index-name* `/_forcemerge` , */index-name* `/update/`*id* y */index-name* `/_close`)
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_ltr`
- `/_mapping`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_resolve/index`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search/pipeline`

- /_bulk
- /_cat (excepto /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings para varias propiedades[4]:
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
  - cluster.max_shards_per_node
  - cluster.search.request.slowlog.level
  - cluster.search.request.slowlog.threshold.warn
  - cluster.search.request.slowlog.threshold.info
  - cluster.search.request.slowlog.threshold.debug
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_plugins/_asynchronous_search
- /_plugins/_alerting
- /_plugins/_anomaly_detection
- /_plugins/_ism
- /_plugins/_ml
- /_plugins/_notifications
- /_plugins/_ppl
- /_plugins/_security
- /_plugins/_security_analytics
- /_plugins/_sm
- /_plugins/_sql
- /_percolate
- /_rank_eval
- /_search/point_in_time
- /_search profile
- /_shard_stores
- /_shrink[5]
- /_snapshot
- /_split
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

    - `cluster.search.request.slowlog.threshold.trace`
    - `search.phase_took_enabled`
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_dashboards`

1. Los cambios en la configuración de un clúster pueden interrumpir estas operaciones antes de que finalicen. Recomendamos utilizar la operación `/_tasks` combinada con estas operaciones para verificar que las solicitudes se hayan llevado a cabo correctamente.
2. Las solicitudes DELETE para `/_search/scroll` con un cuerpo de mensaje deben especificar `"Content-Length"` en el encabezado HTTP. La mayoría de los clientes agregan este encabezado de forma predeterminada. Para evitar problemas con caracteres = en valores `scroll_id`, utilice el cuerpo de la solicitud, no la cadena de solicitud, para pasar valores de `scroll_id` a OpenSearch Service.
3. Consulte para ver consideraciones sobre la utilización de scripts the section called “Otros recursos admitidos”.
4. Se refiere al método `PUT`. Para obtener información acerca del método `GET`, consulte the section called “Diferencias de API destacadas”. Esta lista solo hace referencia a las operaciones genéricas de OpenSearch compatibles con OpenSearch Service y no incluye operaciones compatibles con complementos específicos para detección de anomalías, ISM, etc.
5. Consulte the section called “Reducir”.

OpenSearch versión 2.11

En OpenSearch 2.11, OpenSearch Service admite las siguientes operaciones. Para obtener información sobre la mayoría de las operaciones, consulte OpenSearch REST API reference, o la referencia de la API del complemento específico.

- Todas las operaciones de la ruta de índice (como */index-name* */_forcemerge* , */index-name* */update/id* y */index-name* */_close*)
- /_alias
- /_aliases
- /_all
- /_analyze
- /_bulk
- /_cat (excepto /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings para varias propiedades[4]:
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
  - cluster.max_shards_per_node
- /_cluster/state
- /_cluster/stats
- /_delete_by_query [1]
- /_explain
- /_field_caps
- /_field_stats
- /_flush
- /_ingest/pipeline
- /_ltr
- /_mapping
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_plugins/_asynchronous_search
- /_plugins/_alerting
- /_plugins/_anomaly_detection
- /_plugins/_ism
- /_plugins/_ml
- /_plugins/_notifications
- /_plugins/_ppl
- /_plugins/_security
- /_plugins/_security_analytics
- /_plugins/_sm
- /_plugins/_sql
- /_percolate
- /_rank_eval
- /_refresh
- /_reindex [1]
- /_render
- /_resolve/index
- /_rollover
- /_scripts [3]
- /_search[2]
- /_search/pipeline
- /_search/point_in_time
- /_search profile
- /_shard_stores
- /_shrink[5]
- /_snapshot
- /_split
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

- `/_count`
- `/_dashboards`

1. Los cambios en la configuración de un clúster pueden interrumpir estas operaciones antes de que finalicen. Recomendamos utilizar la operación `/_tasks` combinada con estas operaciones para verificar que las solicitudes se hayan llevado a cabo correctamente.
2. Las solicitudes DELETE para `/_search/scroll` con un cuerpo de mensaje deben especificar `"Content-Length"` en el encabezado HTTP. La mayoría de los clientes agregan este encabezado de forma predeterminada. Para evitar problemas con caracteres = en valores `scroll_id`, utilice el cuerpo de la solicitud, no la cadena de solicitud, para pasar valores de `scroll_id` a OpenSearch Service.
3. Consulte para ver consideraciones sobre la utilización de scripts the section called “Otros recursos admitidos”.
4. Se refiere al método `PUT`. Para obtener información acerca del método `GET`, consulte the section called “Diferencias de API destacadas”. Esta lista solo hace referencia a las operaciones genéricas de OpenSearch compatibles con OpenSearch Service y no incluye operaciones compatibles con complementos específicos para detección de anomalías, ISM, etc.
5. Consulte the section called “Reducir”.

OpenSearch versión 2.9

En OpenSearch 2.9, OpenSearch Service admite las siguientes operaciones. Para obtener información sobre la mayoría de las operaciones, consulte OpenSearch REST API reference, o la referencia de la API del complemento específico.

- Todas las operaciones de la ruta de índice (como */index-name* `/_forcemerge` , */index-name* `/update/`*id* y */index-name* `/_close`)
- `/_alias`
- `/_aliases`
- `/_all`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_ltr`
- `/_mapping`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_resolve/index`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search/pipeline`

- /_analyze
- /_bulk
- /_cat (excepto /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings para varias propiedades[4]:
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
  - cluster.max_shards_per_node
- /_cluster/state
- /_cluster/stats
- /_count
- /_dashboards
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_plugins/_asynchronous_search
- /_plugins/_alerting
- /_plugins/_anomaly_detection
- /_plugins/_ism
- /_plugins/_ml
- /_plugins/_notifications
- /_plugins/_ppl
- /_plugins/_security
- /_plugins/_security_analytics
- /_plugins/_sm
- /_plugins/_sql
- /_percolate
- /_rank_eval
- /_search/point_in_time
- /_search profile
- /_shard_stores
- /_shrink[5]
- /_snapshot
- /_split
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

1. Los cambios en la configuración de un clúster pueden interrumpir estas operaciones antes de que finalicen. Recomendamos utilizar la operación /_tasks combinada con estas operaciones para verificar que las solicitudes se hayan llevado a cabo correctamente.
2. Las solicitudes DELETE para /_search/scroll con un cuerpo de mensaje deben especificar "Content-Length" en el encabezado HTTP. La mayoría de los clientes agregan este

encabezado de forma predeterminada. Para evitar problemas con caracteres = en valores `scroll_id`, utilice el cuerpo de la solicitud, no la cadena de solicitud, para pasar valores de `scroll_id` a OpenSearch Service.

3. Consulte para ver consideraciones sobre la utilización de scripts the section called “Otros recursos admitidos”.
4. Se refiere al método `PUT`. Para obtener información acerca del método `GET`, consulte the section called “Diferencias de API destacadas”. Esta lista solo hace referencia a las operaciones genéricas de OpenSearch compatibles con OpenSearch Service y no incluye operaciones compatibles con complementos específicos para detección de anomalías, ISM, etc.
5. Consulte the section called “Reducir”.

OpenSearch versión 2.7

En OpenSearch 2.7, OpenSearch Service admite las siguientes operaciones. Para obtener información sobre la mayoría de las operaciones, consulte OpenSearch REST API reference, o la referencia de la API del complemento específico.

- Todas las operaciones de la ruta de índice (como */index-name* `/_forcemerge` , */index-name* `/update/`*id* y */index-name* `/_close`)
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (excepto `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_ltr`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_plugins/_asynchronous_search`
- `/_plugins/_alerting`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_resolve/index`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search/point_in_time`
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`
- `/_split`
- `/_stats`

- `/_cluster/settings` para varias propiedades[4]:
  - `action.auto_create_index`
  - `action.search.shard_count.limit`
  - `indices.breaker.fielddata.limit`
  - `indices.breaker.request.limit`
  - `indices.breaker.total.limit`
  - `cluster.max_shards_per_node`
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_dashboards`
- `/_plugins/_anomaly_detection`
- `/_plugins/_ism`
- `/_plugins/_ml`
- `/_plugins/_notifications`
- `/_plugins/_ppl`
- `/_plugins/_security`
- `/_plugins/_security_analytics`
- `/_plugins/_sm`
- `/_plugins/_sql`
- `/_percolate`
- `/_rank_eval`
- `/_status`
- `/_tasks`
- `/_template`
- `/_update_by_query` [1]
- `/_validate`

1. Los cambios en la configuración de un clúster pueden interrumpir estas operaciones antes de que finalicen. Recomendamos utilizar la operación `/_tasks` combinada con estas operaciones para verificar que las solicitudes se hayan llevado a cabo correctamente.
2. Las solicitudes DELETE para `/_search/scroll` con un cuerpo de mensaje deben especificar `"Content-Length"` en el encabezado HTTP. La mayoría de los clientes agregan este encabezado de forma predeterminada. Para evitar problemas con caracteres = en valores `scroll_id`, utilice el cuerpo de la solicitud, no la cadena de solicitud, para pasar valores de `scroll_id` a OpenSearch Service.
3. Consulte para ver consideraciones sobre la utilización de scripts the section called “Otros recursos admitidos”.
4. Se refiere al método `PUT`. Para obtener información acerca del método `GET`, consulte the section called “Diferencias de API destacadas”. Esta lista solo hace referencia a las operaciones genéricas de OpenSearch compatibles con OpenSearch Service y no incluye operaciones compatibles con complementos específicos para detección de anomalías, ISM, etc.

5. Consulte the section called “Reducir”.

OpenSearch versión 2.5

En OpenSearch 2.5, OpenSearch Service admite las siguientes operaciones. Para obtener información sobre la mayoría de las operaciones, consulte OpenSearch REST API reference, o la referencia de la API del complemento específico.

- Todas las operaciones de la ruta de índice (como */index-name* `/_forcemerge` , */index-name* `/update/`*id* y */index-name* `/_close`)
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (excepto `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/settings` para varias propiedades[4]:
  - `action.auto_create_index`
  - `action.search.shard_count.limit`
  - `indices.breaker.fielddata.limit`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_ltr`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_plugins/_asynchronous_search`
- `/_plugins/_alerting`
- `/_plugins/_anomaly_detection`
- `/_plugins/_ism`
- `/_plugins/_ml`
- `/_plugins/_notifications`
- `/_plugins/_ppl`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_resolve/index`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search/point_in_time`
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`
- `/_split`
- `/_stats`
- `/_status`
- `/_tasks`
- `/_template`
- `/_update_by_query` [1]
- `/_validate`

- indices.breaker.request.limit
- indices.breaker.total.limit
- cluster.max_shards_per_node
- /_cluster/state
- /_cluster/stats
- /_count
- /_dashboards
- /_plugins/_security
- /_plugins/_security_analytics
- /_plugins/_sm
- /_plugins/_sql
- /_percolate
- /_rank_eval

1. Los cambios en la configuración de un clúster pueden interrumpir estas operaciones antes de que finalicen. Recomendamos utilizar la operación `/_tasks` combinada con estas operaciones para verificar que las solicitudes se hayan llevado a cabo correctamente.
2. Las solicitudes DELETE para `/_search/scroll` con un cuerpo de mensaje deben especificar `"Content-Length"` en el encabezado HTTP. La mayoría de los clientes agregan este encabezado de forma predeterminada. Para evitar problemas con caracteres = en valores `scroll_id`, utilice el cuerpo de la solicitud, no la cadena de solicitud, para pasar valores de `scroll_id` a OpenSearch Service.
3. Consulte para ver consideraciones sobre la utilización de scripts the section called “Otros recursos admitidos”.
4. Se refiere al método `PUT`. Para obtener información acerca del método `GET`, consulte the section called “Diferencias de API destacadas”. Esta lista solo hace referencia a las operaciones genéricas de OpenSearch compatibles con OpenSearch Service y no incluye operaciones compatibles con complementos específicos para detección de anomalías, ISM, etc.
5. Consulte the section called “Reducir”.

OpenSearch versión 2.3

En OpenSearch 2.3, OpenSearch Service admite las siguientes operaciones. Para obtener información sobre la mayoría de las operaciones, consulte OpenSearch REST API reference, o la referencia de la API del complemento específico.

- Todas las operaciones de la ruta de índice (como */index-name* /_forcemerge , */index-name* /update/*id* y */index-name* /_close)
- /_alias
- /_aliases
- /_all
- /_analyze
- /_bulk
- /_cat (excepto /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings para varias propiedades[4]:
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
  - cluster.max_shards_per_node
- /_cluster/state
- /_cluster/stats
- /_delete_by_query [1]
- /_explain
- /_field_caps
- /_field_stats
- /_flush
- /_ingest/pipeline
- /_ltr
- /_mapping
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_plugins/_asynchronous_search
- /_plugins/_alerting
- /_plugins/_anomaly_detection
- /_plugins/_ism
- /_plugins/_ml
- _plugins/_notifications
- /_plugins/_ppl
- /_plugins/_security
- /_plugins/_sql
- /_percolate
- /_rank_eval
- /_refresh
- /_reindex [1]
- /_render
- /_resolve/index
- /_rollover
- /_scripts [3]
- /_search[2]
- /_search profile
- /_shard_stores
- /_shrink[5]
- /_snapshot
- /_split
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

- `/_count`
- `/_dashboards`

1. Los cambios en la configuración de un clúster pueden interrumpir estas operaciones antes de que finalicen. Recomendamos utilizar la operación `/_tasks` combinada con estas operaciones para verificar que las solicitudes se hayan llevado a cabo correctamente.
2. Las solicitudes DELETE para `/_search/scroll` con un cuerpo de mensaje deben especificar `"Content-Length"` en el encabezado HTTP. La mayoría de los clientes agregan este encabezado de forma predeterminada. Para evitar problemas con caracteres = en valores `scroll_id`, utilice el cuerpo de la solicitud, no la cadena de solicitud, para pasar valores de `scroll_id` a OpenSearch Service.
3. Consulte para ver consideraciones sobre la utilización de scripts the section called “Otros recursos admitidos”.
4. Se refiere al método `PUT`. Para obtener información acerca del método `GET`, consulte the section called “Diferencias de API destacadas”. Esta lista solo hace referencia a las operaciones genéricas de OpenSearch compatibles con OpenSearch Service y no incluye operaciones compatibles con complementos específicos para detección de anomalías, ISM, etc.
5. Consulte the section called “Reducir”.

OpenSearch versión 1.3

En OpenSearch 1.3, OpenSearch Service admite las siguientes operaciones. Para obtener información sobre la mayoría de las operaciones, consulte OpenSearch REST API reference, o la referencia de la API del complemento específico.

- Todas las operaciones de la ruta de índice (como `/index-name /_forcemerge` , `/index-name /update/id` y `/index-name /_close`)
- `/_alias`
- `/_aliases`
- `/_all`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_ltr`
- `/_mapping`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_resolve/index`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search profile`

- /_analyze
- /_bulk
- /_cat (excepto /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings para varias propiedades[4]:
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
  - cluster.max_shards_per_node
- /_cluster/state
- /_cluster/stats
- /_count
- /_dashboards
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_plugins/_asynchronous_search
- /_plugins/_alerting
- /_plugins/_anomaly_detection
- /_plugins/_ism
- /_plugins/_ml
- /_plugins/_ppl
- /_plugins/_security
- /_plugins/_sql
- /_percolate
- /_rank_eval
- /_shard_stores
- /_shrink[5]
- /_snapshot
- /_split
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

1. Los cambios en la configuración de un clúster pueden interrumpir estas operaciones antes de que finalicen. Recomendamos utilizar la operación /_tasks combinada con estas operaciones para verificar que las solicitudes se hayan llevado a cabo correctamente.
2. Las solicitudes DELETE para /_search/scroll con un cuerpo de mensaje deben especificar "Content-Length" en el encabezado HTTP. La mayoría de los clientes agregan este

encabezado de forma predeterminada. Para evitar problemas con caracteres = en valores `scroll_id`, utilice el cuerpo de la solicitud, no la cadena de solicitud, para pasar valores de `scroll_id` a OpenSearch Service.

3. Consulte para ver consideraciones sobre la utilización de scripts the section called “Otros recursos admitidos”.
4. Se refiere al método `PUT`. Para obtener información acerca del método `GET`, consulte the section called “Diferencias de API destacadas”. Esta lista solo hace referencia a las operaciones genéricas de OpenSearch compatibles con OpenSearch Service y no incluye operaciones compatibles con complementos específicos para detección de anomalías, ISM, etc.
5. Consulte the section called “Reducir”.

OpenSearch versión 1.2

En OpenSearch 1.2, OpenSearch Service admite las siguientes operaciones. Para obtener información sobre la mayoría de las operaciones, consulte OpenSearch REST API reference, o la referencia de la API del complemento específico.

- Todas las operaciones de la ruta de índice (como */index-name* `/_forcemerge` , */index-name* `/update/`*id* y */index-name* `/_close`)
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (excepto `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_ltr`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_plugins/_asynchronous_search`
- `/_plugins/_alerting`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_resolve/index`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`
- `/_split`
- `/_stats`
- `/_status`
- `/_tasks`

- `/_cluster/settings` para varias propiedades[4]:
  - `action.auto_create_index`
  - `action.search.shard_count.limit`
  - `indices.breaker.fielddata.limit`
  - `indices.breaker.request.limit`
  - `indices.breaker.total.limit`
  - `cluster.max_shards_per_node`
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_dashboards`
- `/_plugins/_anomaly_detection`
- `/_plugins/_ism`
- `/_plugins/_ppl`
- `/_plugins/_security`
- `/_plugins/_sql`
- `/_percolate`
- `/_rank_eval`
- `/_template`
- `/_update_by_query` [1]
- `/_validate`

1. Los cambios en la configuración de un clúster pueden interrumpir estas operaciones antes de que finalicen. Recomendamos utilizar la operación `/_tasks` combinada con estas operaciones para verificar que las solicitudes se hayan llevado a cabo correctamente.
2. Las solicitudes DELETE para `/_search/scroll` con un cuerpo de mensaje deben especificar `"Content-Length"` en el encabezado HTTP. La mayoría de los clientes agregan este encabezado de forma predeterminada. Para evitar problemas con caracteres = en valores `scroll_id`, utilice el cuerpo de la solicitud, no la cadena de solicitud, para pasar valores de `scroll_id` a OpenSearch Service.
3. Consulte para ver consideraciones sobre la utilización de scripts the section called “Otros recursos admitidos”.
4. Se refiere al método `PUT`. Para obtener información acerca del método `GET`, consulte the section called “Diferencias de API destacadas”. Esta lista solo hace referencia a las operaciones genéricas de OpenSearch compatibles con OpenSearch Service y no incluye operaciones compatibles con complementos específicos para detección de anomalías, ISM, etc.

5. Consulte the section called “Reducir”.

OpenSearch versión 1.1

En OpenSearch 1.1, OpenSearch Service es compatible con las siguientes operaciones. Para obtener información sobre la mayoría de las operaciones, consulte OpenSearch REST API reference, o la referencia de la API del complemento específico.

- Todas las operaciones de la ruta de índice (como */index-name* `/_forcemerge` , */index-name* `/update/`*id* y */index-name* `/_close`)
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (excepto `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/settings` para varias propiedades[4]:
  - `action.auto_create_index`
  - `action.search.shard_count.limit`
  - `indices.breaker.fielddata.limit`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_ltr`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_plugins/_asynchronous_search`
- `/_plugins/_alerting`
- `/_plugins/_anomaly_detection`
- `/_plugins/_ism`
- `/_plugins/_ppl`
- `/_plugins/_security`
- `/_plugins/_sql`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_resolve/index`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`
- `/_split`
- `/_stats`
- `/_status`
- `/_tasks`
- `/_template`
- `/_update_by_query` [1]
- `/_validate`

  - indices.breaker.request.limit
  - indices.breaker.total.limit
  - cluster.max_shards_per_node
- /_cluster/state
- /_cluster/stats
- /_count
- /_dashboards
- /_plugins/_transforms
- /_percolate
- /_rank_eval

1. Los cambios en la configuración de un clúster pueden interrumpir estas operaciones antes de que finalicen. Recomendamos utilizar la operación /_tasks combinada con estas operaciones para verificar que las solicitudes se hayan llevado a cabo correctamente.
2. Las solicitudes DELETE para /_search/scroll con un cuerpo de mensaje deben especificar "Content-Length" en el encabezado HTTP. La mayoría de los clientes agregan este encabezado de forma predeterminada. Para evitar problemas con caracteres = en valores scroll_id, utilice el cuerpo de la solicitud, no la cadena de solicitud, para pasar valores de scroll_id a OpenSearch Service.
3. Consulte para ver consideraciones sobre la utilización de scripts the section called “Otros recursos admitidos”.
4. Se refiere al método PUT. Para obtener información acerca del método GET, consulte the section called “Diferencias de API destacadas”. Esta lista solo hace referencia a las operaciones genéricas de OpenSearch compatibles con OpenSearch Service y no incluye operaciones compatibles con complementos específicos para detección de anomalías, ISM, etc.
5. Consulte the section called “Reducir”.

OpenSearch versión 1.0

En OpenSearch 1.0, OpenSearch Service es compatible con las siguientes operaciones. Para obtener información sobre la mayoría de las operaciones, consulte OpenSearch REST API reference, o la referencia de la API del complemento específico.

- Todas las operaciones de la ruta de índice (como */index-name* */_forcemerge* , */index-name* */update/id* y */index-name* */_close*)
- /_alias
- /_aliases
- /_all
- /_analyze
- /_bulk
- /_cat (excepto /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings para varias propiedades[4]:
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
  - cluster.max_shards_per_node
- /_cluster/state
- /_cluster/stats
- /_delete_by_query [1]
- /_explain
- /_field_caps
- /_field_stats
- /_flush
- /_ingest/pipeline
- /_ltr
- /_mapping
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_plugins/_asynchronous_search
- /_plugins/_alerting
- /_plugins/_anomaly_detection
- /_plugins/_ism
- /_plugins/_ppl
- /_plugins/_security
- /_plugins/_sql
- /_plugins/_transforms
- /_percolate
- /_rank_eval
- /_refresh
- /_reindex [1]
- /_render
- /_resolve/index
- /_rollover
- /_scripts [3]
- /_search[2]
- /_search profile
- /_shard_stores
- /_shrink[5]
- /_snapshot
- /_split
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

- `/_count`
- `/_dashboards`

1. Los cambios en la configuración de un clúster pueden interrumpir estas operaciones antes de que finalicen. Recomendamos utilizar la operación `/_tasks` combinada con estas operaciones para verificar que las solicitudes se hayan llevado a cabo correctamente.
2. Las solicitudes DELETE para `/_search/scroll` con un cuerpo de mensaje deben especificar `"Content-Length"` en el encabezado HTTP. La mayoría de los clientes agregan este encabezado de forma predeterminada. Para evitar problemas con caracteres = en valores `scroll_id`, utilice el cuerpo de la solicitud, no la cadena de solicitud, para pasar valores de `scroll_id` a OpenSearch Service.
3. Consulte para ver consideraciones sobre la utilización de scripts the section called “Otros recursos admitidos”.
4. Se refiere al método `PUT`. Para obtener información acerca del método `GET`, consulte the section called “Diferencias de API destacadas”. Esta lista solo hace referencia a las operaciones genéricas de OpenSearch compatibles con OpenSearch Service y no incluye operaciones compatibles con complementos específicos para detección de anomalías, ISM, etc.
5. Consulte the section called “Reducir”.

Elasticsearch versión 7.10

En Elasticsearch 7.10, OpenSearch Service es compatible con las siguientes operaciones.

- Todas las operaciones de la ruta de índice (como */index-name* `/_forcemerge` , */index-name* `/update/`*id* y */index-name* `/_close`)
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_index_template` [6]
- `/_ingest/pipeline`
- `/_index_template`
- `/_ltr`
- `/_mapping`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_resolve/index`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]

- /_cat (excepto /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings para varias propiedades[4]:
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
  - cluster.max_shards_per_node
- /_cluster/state
- /_cluster/stats
- /_count
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_opendistro/_alerting
- /_opendistro/_asynchronous_search
- /_opendistro/_anomaly_detection
- /_opendistro/_ism
- /_opendistro/_ppl
- /_opendistro/_security
- /_opendistro/_sql
- /_percolate
- /_plugin/kibana
- /_plugins/_replication
- /_rank_eval
- /_snapshot
- /_split
- /_stats
- /_status
- /_tasks
- /_template [6]
- /_update_by_query [1]
- /_validate

1. Los cambios en la configuración de un clúster pueden interrumpir estas operaciones antes de que finalicen. Recomendamos utilizar la operación /_tasks combinada con estas operaciones para verificar que las solicitudes se hayan llevado a cabo correctamente.
2. Las solicitudes DELETE para /_search/scroll con un cuerpo de mensaje deben especificar "Content-Length" en el encabezado HTTP. La mayoría de los clientes agregan este encabezado de forma predeterminada. Para evitar problemas con caracteres = en valores scroll_id, utilice el cuerpo de la solicitud, no la cadena de solicitud, para pasar valores de scroll_id a OpenSearch Service.

3. Consulte para ver consideraciones sobre la utilización de scripts the section called “Otros recursos admitidos”.
4. Se refiere al método PUT. Para obtener información acerca del método GET, consulte the section called “Diferencias de API destacadas”. Esta lista solo hace referencia a las operaciones genéricas de Elasticsearch que OpenSearch Service admite y no incluye operaciones compatibles con complementos específicos para detección de anomalías, ISM, etc.
5. Consulte the section called “Reducir”.
6. Las plantillas de índice heredadas (`_template`) fueron reemplazadas por plantillas que se pueden combinar (`_index_template`), a partir de Elasticsearch 7.8. Las plantillas que se pueden combinar tienen prioridad sobre las plantillas heredadas. Si ninguna plantilla que se puede combinar coincide con un índice determinado, una plantilla heredada puede coincidir y ser aplicada. La operación `_template` aún funciona en OpenSearch y versiones posteriores de Elasticsearch OSS, pero las llamadas GET a los dos tipos de plantillas devuelven resultados diferentes.

Elasticsearch versión 7.9

En Elasticsearch 7.9, OpenSearch Service es compatible con las siguientes operaciones.

- Todas las operaciones de la ruta de índice (como */index-name* `/_forcemerge` , */index-name* `/update/`*id* y */index-name* `/_close`)
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (excepto `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_index_template` [6]
- `/_ingest/pipeline`
- `/_ltr`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_resolve/index`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`
- `/_split`
- `/_stats`
- `/_status`

- `/_cluster/pending_tasks`
- `/_cluster/settings` para varias propiedades[4]:
  - `action.auto_create_index`
  - `action.search.shard_count.limit`
  - `indices.breaker.fielddata.limit`
  - `indices.breaker.request.limit`
  - `indices.breaker.total.limit`
  - `cluster.max_shards_per_node`
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_opendistro/_alerting`
- `/_opendistro/_anomaly_detection`
- `/_opendistro/_ism`
- `/_opendistro/_ppl`
- `/_opendistro/_security`
- `/_opendistro/_sql`
- `/_percolate`
- `/_plugin/kibana`
- `/_rank_eval`
- `/_tasks`
- `/_template` [6]
- `/_update_by_query` [1]
- `/_validate`

1. Los cambios en la configuración de un clúster pueden interrumpir estas operaciones antes de que finalicen. Recomendamos utilizar la operación `/_tasks` combinada con estas operaciones para verificar que las solicitudes se hayan llevado a cabo correctamente.
2. Las solicitudes DELETE para `/_search/scroll` con un cuerpo de mensaje deben especificar `"Content-Length"` en el encabezado HTTP. La mayoría de los clientes agregan este encabezado de forma predeterminada. Para evitar problemas con caracteres = en valores `scroll_id`, utilice el cuerpo de la solicitud, no la cadena de solicitud, para pasar valores de `scroll_id` a OpenSearch Service.
3. Consulte para ver consideraciones sobre la utilización de scripts the section called “Otros recursos admitidos”.
4. Se refiere al método `PUT`. Para obtener información acerca del método `GET`, consulte the section called “Diferencias de API destacadas”. Esta lista solo hace referencia a las operaciones genéricas de OpenSearch compatibles con OpenSearch Service y no incluye operaciones compatibles con complementos específicos para detección de anomalías, ISM, etc.

5. Consulte the section called “Reducir”.
6. Las plantillas de índice heredadas (`_template`) fueron reemplazadas por plantillas que se pueden combinar (`_index_template`), a partir de Elasticsearch 7.8. Las plantillas que se pueden combinar tienen prioridad sobre las plantillas heredadas. Si ninguna plantilla que se puede combinar coincide con un índice determinado, una plantilla heredada puede coincidir y ser aplicada. La operación `_template` aún funciona en OpenSearch y versiones posteriores de Elasticsearch OSS, pero las llamadas GET a los dos tipos de plantillas devuelven resultados diferentes.

Elasticsearch versión 7.8

En Elasticsearch 7.8, OpenSearch Service admite las siguientes operaciones.

- Todas las operaciones de la ruta de índice (como */index-name* `/_forcemerge` , */index-name* `/update/`*id* y */index-name* `/_close`)
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (excepto `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/settings` para varias propiedades[4]:
  - `action.auto_create_index`
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_index_template` [6]
- `/_ingest/pipeline`
- `/_ltr`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_opendistro/_alerting`
- `/_opendistro/_anomaly_detection`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`
- `/_split`
- `/_stats`
- `/_status`
- `/_tasks`
- `/_template` [6]
- `/_update_by_query` [1]
- `/_validate`

- `action.search.shard_count.limit`
- `indices.breaker.fielddata.limit`
- `indices.breaker.request.limit`
- `indices.breaker.total.limit`
- `cluster.max_shards_per_node`
- `/_opendistro/_ism`
- `/_opendistro/_security`
- `/_opendistro/_sql`
- `/_percolate`
- `/_plugin/kibana`
- `/_rank_eval`

1. Los cambios en la configuración de un clúster pueden interrumpir estas operaciones antes de que finalicen. Recomendamos utilizar la operación `/_tasks` combinada con estas operaciones para verificar que las solicitudes se hayan llevado a cabo correctamente.
2. Las solicitudes DELETE para `/_search/scroll` con un cuerpo de mensaje deben especificar `"Content-Length"` en el encabezado HTTP. La mayoría de los clientes agregan este encabezado de forma predeterminada. Para evitar problemas con caracteres = en valores `scroll_id`, utilice el cuerpo de la solicitud, no la cadena de solicitud, para pasar valores de `scroll_id` a OpenSearch Service.
3. Consulte para ver consideraciones sobre la utilización de scripts the section called “Otros recursos admitidos”.
4. Se refiere al método `PUT`. Para obtener información acerca del método `GET`, consulte the section called “Diferencias de API destacadas”. Esta lista solo hace referencia a las operaciones genéricas de Elasticsearch que OpenSearch Service admite y no incluye operaciones compatibles con complementos específicos para detección de anomalías, ISM, etc.
5. Consulte the section called “Reducir”.
6. Las plantillas de índice heredadas (`_template`) fueron reemplazadas por plantillas que se pueden combinar (`_index_template`), a partir de Elasticsearch 7.8. Las plantillas que se pueden combinar tienen prioridad sobre las plantillas heredadas. Si ninguna plantilla que se puede combinar coincide con un índice determinado, una plantilla heredada puede coincidir y ser aplicada. La operación `_template` aún funciona en OpenSearch y versiones posteriores de Elasticsearch OSS, pero las llamadas GET a los dos tipos de plantillas devuelven resultados diferentes.

Elasticsearch versión 7.7

En Elasticsearch 7.7, OpenSearch Service admite las siguientes operaciones.

- Todas las operaciones de la ruta de índice (como `/index-name /_forcemerge` , `/index-name /update/id` y `/index-name /_close`)
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (excepto `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/settings` para varias propiedades[4]:
  - `action.auto_create_index`
  - `action.search.shard_count.limit`
  - `indices.breaker.fielddata.limit`
  - `indices.breaker.request.limit`
  - `indices.breaker.total.limit`
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_ltr`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_opendistro/_alerting`
- `/_opendistro/_anomaly_detection`
- `/_opendistro/_ism`
- `/_opendistro/_security`
- `/_opendistro/_sql`
- `/_percolate`
- `/_plugin/kibana`
- `/_rank_eval`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`
- `/_split`
- `/_stats`
- `/_status`
- `/_tasks`
- `/_template`
- `/_update_by_query` [1]
- `/_validate`

- `cluster.max_shards_per_node`

1. Los cambios en la configuración de un clúster pueden interrumpir estas operaciones antes de que finalicen. Recomendamos utilizar la operación `/_tasks` combinada con estas operaciones para verificar que las solicitudes se hayan llevado a cabo correctamente.
2. Las solicitudes DELETE para `/_search/scroll` con un cuerpo de mensaje deben especificar `"Content-Length"` en el encabezado HTTP. La mayoría de los clientes agregan este encabezado de forma predeterminada. Para evitar problemas con caracteres = en valores `scroll_id`, utilice el cuerpo de la solicitud, no la cadena de solicitud, para pasar valores de `scroll_id` a OpenSearch Service.
3. Consulte para ver consideraciones sobre la utilización de scripts the section called “Otros recursos admitidos”.
4. Se refiere al método `PUT`. Para obtener información acerca del método `GET`, consulte the section called “Diferencias de API destacadas”. Esta lista solo hace referencia a las operaciones genéricas de Elasticsearch que OpenSearch Service admite y no incluye operaciones compatibles con complementos específicos para detección de anomalías, ISM, etc.
5. Consulte the section called “Reducir”.

Elasticsearch versión 7.4

En Elasticsearch 7.4, OpenSearch Service admite las siguientes operaciones.

- Todas las operaciones de la ruta de índice (como */index-name* `/_forcemerge` , */index-name* `/update/`*id* y */index-name* `/_close`)
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_mapping`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`

- `/_cat` (excepto `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/settings` para varias propiedades[4]:
  - `action.auto_create_index`
  - `action.search.shard_count.limit`
  - `indices.breaker.fielddata.limit`
  - `indices.breaker.request.limit`
  - `indices.breaker.total.limit`
  - `cluster.max_shards_per_node`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_opendistro/_alerting`
- `/_opendistro/_anomaly_detection`
- `/_opendistro/_ism`
- `/_opendistro/_security`
- `/_opendistro/_sql`
- `/_percolate`
- `/_plugin/kibana`
- `/_rank_eval`
- `/_split`
- `/_stats`
- `/_status`
- `/_tasks`
- `/_template`
- `/_update_by_query` [1]
- `/_validate`

1. Los cambios en la configuración de un clúster pueden interrumpir estas operaciones antes de que finalicen. Recomendamos utilizar la operación `/_tasks` combinada con estas operaciones para verificar que las solicitudes se hayan llevado a cabo correctamente.
2. Las solicitudes DELETE para `/_search/scroll` con un cuerpo de mensaje deben especificar `"Content-Length"` en el encabezado HTTP. La mayoría de los clientes agregan este encabezado de forma predeterminada. Para evitar problemas con caracteres = en valores `scroll_id`, utilice el cuerpo de la solicitud, no la cadena de solicitud, para pasar valores de `scroll_id` a OpenSearch Service.
3. Consulte para ver consideraciones sobre la utilización de scripts the section called “Otros recursos admitidos”.
4. Se refiere al método `PUT`. Para obtener información acerca del método `GET`, consulte the section called “Diferencias de API destacadas”. Esta lista solo hace referencia a las operaciones

genéricas de Elasticsearch que OpenSearch Service admite y no incluye operaciones compatibles con complementos específicos para detección de anomalías, ISM, etc.

5. Consulte the section called “Reducir”.

Elasticsearch versión 7.1

En Elasticsearch 7.1, OpenSearch Service admite las siguientes operaciones.

- Todas las operaciones en la ruta de índice (como */index-name* `/_forcemerge` y */index-name* `/update/`*id*) excepto */index-name* `/_close`
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (excepto `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/settings` para varias propiedades[4]:
  - `action.auto_create_index`
  - `action.search.shard_count.limit`
  - `indices.breaker.fielddata.limit`
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_opendistro/_alerting`
- `/_opendistro/_ism`
- `/_opendistro/_security`
- `/_opendistro/_sql`
- `/_percolate`
- `/_plugin/kibana`
- `/_rank_eval`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`
- `/_split`
- `/_stats`
- `/_status`
- `/_tasks`
- `/_template`
- `/_update_by_query` [1]
- `/_validate`

- indices.breaker.request.limit
- indices.breaker.total.limit
- cluster.max_shards_per_node

1. Los cambios en la configuración de un clúster pueden interrumpir estas operaciones antes de que finalicen. Recomendamos utilizar la operación `/_tasks` combinada con estas operaciones para verificar que las solicitudes se hayan llevado a cabo correctamente.
2. Las solicitudes DELETE para `/_search/scroll` con un cuerpo de mensaje deben especificar `"Content-Length"` en el encabezado HTTP. La mayoría de los clientes agregan este encabezado de forma predeterminada. Para evitar problemas con caracteres = en valores `scroll_id`, utilice el cuerpo de la solicitud, no la cadena de solicitud, para pasar valores de `scroll_id` a OpenSearch Service.
3. Consulte para ver consideraciones sobre la utilización de scripts the section called “Otros recursos admitidos”.
4. Se refiere al método `PUT`. Para obtener información acerca del método `GET`, consulte the section called “Diferencias de API destacadas”. Esta lista solo hace referencia a las operaciones genéricas de Elasticsearch que OpenSearch Service admite y no incluye operaciones compatibles con complementos específicos para detección de anomalías, ISM, etc.
5. Consulte the section called “Reducir”.

Elasticsearch versión 6.8

En Elasticsearch 6.8, OpenSearch Service admite las siguientes operaciones.

- Todas las operaciones en la ruta de índice (como */index-name* `/_forcemerge` y */index-name* `/update/`*id*) excepto */index-name* `/_close`
- `/_alias`
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_rollover`
- `/_scripts` [3]
- `/_search` [2]

- /_aliases
- /_all
- /_analyze
- /_bulk
- /_cat (excepto /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings para varias propiedades[4]:
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
  - cluster.max_shards_per_node
  - cluster.blocks.read_only
- /_field_stats
- /_flush
- /_ingest/pipeline
- /_mapping
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_opendistro/_alerting
- /_opendistro/_ism
- /_opendistro/_security
- /_opendistro/_sql
- /_percolate
- /_plugin/kibana
- /_rank_eval
- /_search profile
- /_shard_stores
- /_shrink[5]
- /_snapshot
- /_split
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

1. Los cambios en la configuración de un clúster pueden interrumpir estas operaciones antes de que finalicen. Recomendamos utilizar la operación /_tasks combinada con estas operaciones para verificar que las solicitudes se hayan llevado a cabo correctamente.
2. Las solicitudes DELETE para /_search/scroll con un cuerpo de mensaje deben especificar "Content-Length" en el encabezado HTTP. La mayoría de los clientes agregan este

encabezado de forma predeterminada. Para evitar problemas con caracteres = en valores `scroll_id`, utilice el cuerpo de la solicitud, no la cadena de solicitud, para pasar valores de `scroll_id` a OpenSearch Service.

3. Consulte para ver consideraciones sobre la utilización de scripts the section called “Otros recursos admitidos”.
4. Se refiere al método `PUT`. Para obtener información acerca del método `GET`, consulte the section called “Diferencias de API destacadas”. Esta lista solo hace referencia a las operaciones genéricas de Elasticsearch que OpenSearch Service admite y no incluye operaciones compatibles con complementos específicos para detección de anomalías, ISM, etc.
5. Consulte the section called “Reducir”.

Elasticsearch versión 6.7

En Elasticsearch 6.7, OpenSearch Service admite las siguientes operaciones.

- Todas las operaciones en la ruta de índice (como */index-name* `/_forcemerge` y */index-name* `/update/`*id*) excepto */index-name* `/_close`
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (excepto `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/settings` para varias propiedades[4]:
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_opendistro/_alerting`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`
- `/_split`
- `/_stats`
- `/_status`
- `/_tasks`
- `/_template`
- `/_update_by_query` [1]
- `/_validate`

- `action.auto_create_index`
- `action.search.shard_count.limit`
- `indices.breaker.fielddata.limit`
- `indices.breaker.request.limit`
- `indices.breaker.total.limit`
- `cluster.max_shards_per_node`

- `/_opendistro/_security`
- `/_opendistro/_sql`
- `/_percolate`
- `/_plugin/kibana`
- `/_rank_eval`

1. Los cambios en la configuración de un clúster pueden interrumpir estas operaciones antes de que finalicen. Recomendamos utilizar la operación `/_tasks` combinada con estas operaciones para verificar que las solicitudes se hayan llevado a cabo correctamente.
2. Las solicitudes DELETE para `/_search/scroll` con un cuerpo de mensaje deben especificar `"Content-Length"` en el encabezado HTTP. La mayoría de los clientes agregan este encabezado de forma predeterminada. Para evitar problemas con caracteres = en valores `scroll_id`, utilice el cuerpo de la solicitud, no la cadena de solicitud, para pasar valores de `scroll_id` a OpenSearch Service.
3. Consulte para ver consideraciones sobre la utilización de scripts the section called “Otros recursos admitidos”.
4. Se refiere al método `PUT`. Para obtener información acerca del método `GET`, consulte the section called “Diferencias de API destacadas”. Esta lista solo hace referencia a las operaciones genéricas de Elasticsearch que OpenSearch Service admite y no incluye operaciones compatibles con complementos específicos para detección de anomalías, ISM, etc.
5. Consulte the section called “Reducir”.

Elasticsearch versión 6.5

En Elasticsearch 6.5, OpenSearch Service admite las siguientes operaciones.

- Todas las operaciones en la ruta de índice (como */index-name* `/_forcemerge` y */index-name* `/update/`*id*) excepto */index-name* `/_close`
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (excepto `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/settings` para varias propiedades[4]:
  - `action.auto_create_index`
  - `action.search.shard_count.limit`
  - `indices.breaker.fielddata.limit`
  - `indices.breaker.request.limit`
  - `indices.breaker.total.limit`
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_opendistro/_alerting`
- `/_opendistro/_sql`
- `/_percolate`
- `/_plugin/kibana`
- `/_rank_eval`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`
- `/_split`
- `/_stats`
- `/_status`
- `/_tasks`
- `/_template`
- `/_update_by_query` [1]
- `/_validate`

1. Los cambios en la configuración de un clúster pueden interrumpir estas operaciones antes de que finalicen. Recomendamos utilizar la operación `/_tasks` combinada con estas operaciones para verificar que las solicitudes se hayan llevado a cabo correctamente.

2. Las solicitudes DELETE para `/_search/scroll` con un cuerpo de mensaje deben especificar `"Content-Length"` en el encabezado HTTP. La mayoría de los clientes agregan este encabezado de forma predeterminada. Para evitar problemas con caracteres = en valores `scroll_id`, utilice el cuerpo de la solicitud, no la cadena de solicitud, para pasar valores de `scroll_id` a OpenSearch Service.
3. Consulte para ver consideraciones sobre la utilización de scripts the section called “Otros recursos admitidos”.
4. Se refiere al método `PUT`. Para obtener información acerca del método `GET`, consulte the section called “Diferencias de API destacadas”. Esta lista solo hace referencia a las operaciones genéricas de Elasticsearch que OpenSearch Service admite y no incluye operaciones compatibles con complementos específicos para detección de anomalías, ISM, etc.
5. Consulte the section called “Reducir”.

Elasticsearch versión 6.4

En Elasticsearch 6.4, OpenSearch Service admite las siguientes operaciones.

- Todas las operaciones en la ruta de índice (como */index-name* `/_forcemerge` y */index-name* `/update/`*id*) excepto */index-name* `/_close`
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (excepto `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_opendistro/_alerting`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`
- `/_split`
- `/_stats`
- `/_status`
- `/_tasks`
- `/_template`

- `/_cluster/settings` para varias propiedades[4]:
  - `action.auto_create_index`
  - `action.search.shard_count.limit`
  - `indices.breaker.fielddata.limit`
  - `indices.breaker.request.limit`
  - `indices.breaker.total.limit`
- `/_percolate`
- `/_plugin/kibana`
- `/_rank_eval`
- `/_update_by_query` [1]
- `/_validate`

1. Los cambios en la configuración de un clúster pueden interrumpir estas operaciones antes de que finalicen. Recomendamos utilizar la operación `/_tasks` combinada con estas operaciones para verificar que las solicitudes se hayan llevado a cabo correctamente.
2. Las solicitudes DELETE para `/_search/scroll` con un cuerpo de mensaje deben especificar `"Content-Length"` en el encabezado HTTP. La mayoría de los clientes agregan este encabezado de forma predeterminada. Para evitar problemas con caracteres = en valores `scroll_id`, utilice el cuerpo de la solicitud, no la cadena de solicitud, para pasar valores de `scroll_id` a OpenSearch Service.
3. Consulte para ver consideraciones sobre la utilización de scripts the section called “Otros recursos admitidos”.
4. Se refiere al método `PUT`. Para obtener información acerca del método `GET`, consulte the section called “Diferencias de API destacadas”. Esta lista solo hace referencia a las operaciones genéricas de Elasticsearch que OpenSearch Service admite y no incluye operaciones compatibles con complementos específicos para detección de anomalías, ISM, etc.
5. Consulte the section called “Reducir”.

Elasticsearch versión 6.3

En Elasticsearch 6.3, OpenSearch Service admite las siguientes operaciones.

- Todas las operaciones en la ruta de índice (como */index-name* */_forcemerge* y */index-name* */update/id*) excepto */index-name* */_close*
- */_alias*
- */_aliases*
- */_all*
- */_analyze*
- */_bulk*
- */_cat* (excepto */_cat/nodeattrs*)
- */_cluster/allocation/explain*
- */_cluster/health*
- */_cluster/pending_tasks*
- */_cluster/settings* para varias propiedades[4]:
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
- */_cluster/state*
- */_cluster/stats*
- */_count*
- */_delete_by_query* [1]
- */_explain*
- */_field_caps*
- */_field_stats*
- */_flush*
- */_ingest/pipeline*
- */_mapping*
- */_mget*
- */_msearch*
- */_mtermvectors*
- */_nodes*
- */_opendistro/_alerting*
- */_percolate*
- */_plugin/kibana*
- */_rank_eval*
- */_refresh*
- */_reindex* [1]
- */_render*
- */_rollover*
- */_scripts* [3]
- */_search*[2]
- */_search profile*
- */_shard_stores*
- */_shrink*[5]
- */_snapshot*
- */_split*
- */_stats*
- */_status*
- */_tasks*
- */_template*
- */_update_by_query* [1]
- */_validate*

1. Los cambios en la configuración de un clúster pueden interrumpir estas operaciones antes de que finalicen. Recomendamos utilizar la operación */_tasks* combinada con estas operaciones para verificar que las solicitudes se hayan llevado a cabo correctamente.

2. Las solicitudes DELETE para `/_search/scroll` con un cuerpo de mensaje deben especificar `"Content-Length"` en el encabezado HTTP. La mayoría de los clientes agregan este encabezado de forma predeterminada. Para evitar problemas con caracteres = en valores `scroll_id`, utilice el cuerpo de la solicitud, no la cadena de solicitud, para pasar valores de `scroll_id` a OpenSearch Service.
3. Consulte para ver consideraciones sobre la utilización de scripts the section called “Otros recursos admitidos”.
4. Se refiere al método `PUT`. Para obtener información acerca del método `GET`, consulte the section called “Diferencias de API destacadas”. Esta lista solo hace referencia a las operaciones genéricas de Elasticsearch que OpenSearch Service admite y no incluye operaciones compatibles con complementos específicos para detección de anomalías, ISM, etc.
5. Consulte the section called “Reducir”.

Elasticsearch versión 6.2

En Elasticsearch 6.2, OpenSearch Service admite las siguientes operaciones.

- Todas las operaciones en la ruta de índice (como */index-name* `/_forcemerge` y */index-name* `/update/`*id*) excepto */index-name* `/_close`
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (excepto `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_opendistro/_alerting`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`
- `/_split`
- `/_stats`
- `/_status`
- `/_tasks`
- `/_template`

- `/_cluster/settings` para varias propiedades[4]:
  - `action.auto_create_index`
  - `action.search.shard_count.limit`
  - `indices.breaker.fielddata.limit`
  - `indices.breaker.request.limit`
  - `indices.breaker.total.limit`
- `/_percolate`
- `/_plugin/kibana`
- `/_rank_eval`
- `/_update_by_query` [1]
- `/_validate`

1. Los cambios en la configuración de un clúster pueden interrumpir estas operaciones antes de que finalicen. Recomendamos utilizar la operación `/_tasks` combinada con estas operaciones para verificar que las solicitudes se hayan llevado a cabo correctamente.
2. Las solicitudes DELETE para `/_search/scroll` con un cuerpo de mensaje deben especificar `"Content-Length"` en el encabezado HTTP. La mayoría de los clientes agregan este encabezado de forma predeterminada. Para evitar problemas con caracteres = en valores `scroll_id`, utilice el cuerpo de la solicitud, no la cadena de solicitud, para pasar valores de `scroll_id` a OpenSearch Service.
3. Consulte para ver consideraciones sobre la utilización de scripts the section called “Otros recursos admitidos”.
4. Se refiere al método `PUT`. Para obtener información acerca del método `GET`, consulte the section called “Diferencias de API destacadas”. Esta lista solo hace referencia a las operaciones genéricas de Elasticsearch que OpenSearch Service admite y no incluye operaciones compatibles con complementos específicos para detección de anomalías, ISM, etc.
5. Consulte the section called “Reducir”.

Elasticsearch versión 6.0

En Elasticsearch 6.0, OpenSearch Service admite las siguientes operaciones.

- Todas las operaciones en la ruta de índice (como */index-name* */_forcemerge* y */index-name* */update/id*) excepto */index-name* */_close*
- */_alias*
- */_aliases*
- */_all*
- */_analyze*
- */_bulk*
- */_cat* (excepto */_cat/nodeattrs*)
- */_cluster/allocation/explain*
- */_cluster/health*
- */_cluster/pending_tasks*
- */_cluster/settings* para varias propiedades[4]:
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
- */_cluster/state*
- */_cluster/stats*
- */_count*
- */_delete_by_query* [1]
- */_explain*
- */_field_caps*
- */_field_stats*
- */_flush*
- */_ingest/pipeline*
- */_mapping*
- */_mget*
- */_msearch*
- */_mtermvectors*
- */_nodes*
- */_percolate*
- */_plugin/kibana*
- */_refresh*
- */_reindex* [1]
- */_render*
- */_rollover*
- */_scripts* [3]
- */_search*[2]
- */_search profile*
- */_shard_stores*
- */_shrink*[5]
- */_snapshot*
- */_stats*
- */_status*
- */_tasks*
- */_template*
- */_update_by_query* [1]
- */_validate*

1. Los cambios en la configuración de un clúster pueden interrumpir estas operaciones antes de que finalicen. Recomendamos utilizar la operación */_tasks* combinada con estas operaciones para verificar que las solicitudes se hayan llevado a cabo correctamente.

2. Las solicitudes DELETE para `/_search/scroll` con un cuerpo de mensaje deben especificar `"Content-Length"` en el encabezado HTTP. La mayoría de los clientes agregan este encabezado de forma predeterminada. Para evitar problemas con caracteres = en valores `scroll_id`, utilice el cuerpo de la solicitud, no la cadena de solicitud, para pasar valores de `scroll_id` a OpenSearch Service.
3. Consulte para ver consideraciones sobre la utilización de scripts the section called “Otros recursos admitidos”.
4. Se refiere al método `PUT`. Para obtener información acerca del método `GET`, consulte the section called “Diferencias de API destacadas”. Esta lista solo hace referencia a las operaciones genéricas de Elasticsearch que OpenSearch Service admite y no incluye operaciones compatibles con complementos específicos para detección de anomalías, ISM, etc.
5. Consulte the section called “Reducir”.

Elasticsearch versión 5.6

En Elasticsearch 5.6, OpenSearch Service admite las siguientes operaciones.

- Todas las operaciones en la ruta de índice (como */index-name* `/_forcemerge` y */index-name* `/update/`*id*) excepto */index-name* `/_close`
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (excepto `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_percolate`
- `/_plugin/kibana`
- `/_render`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`
- `/_stats`
- `/_status`
- `/_tasks`
- `/_template`
- `/_update_by_query` [1]
- `/_validate`

- `/_cluster/settings` para varias propiedades[4]:
  - `action.auto_create_index`
  - `action.search.shard_count.limit`
  - `indices.breaker.fielddata.limit`
  - `indices.breaker.request.limit`
  - `indices.breaker.total.limit`
- `/_refresh`
- `/_reindex` [1]

1. Los cambios en la configuración de un clúster pueden interrumpir estas operaciones antes de que finalicen. Recomendamos utilizar la operación `/_tasks` combinada con estas operaciones para verificar que las solicitudes se hayan llevado a cabo correctamente.
2. Las solicitudes DELETE para `/_search/scroll` con un cuerpo de mensaje deben especificar `"Content-Length"` en el encabezado HTTP. La mayoría de los clientes agregan este encabezado de forma predeterminada. Para evitar problemas con caracteres = en valores `scroll_id`, utilice el cuerpo de la solicitud, no la cadena de solicitud, para pasar valores de `scroll_id` a OpenSearch Service.
3. Consulte para ver consideraciones sobre la utilización de scripts the section called “Otros recursos admitidos”.
4. Se refiere al método `PUT`. Para obtener información acerca del método `GET`, consulte the section called “Diferencias de API destacadas”. Esta lista solo hace referencia a las operaciones genéricas de Elasticsearch que OpenSearch Service admite y no incluye operaciones compatibles con complementos específicos para detección de anomalías, ISM, etc.
5. Consulte the section called “Reducir”.

Elasticsearch versión 5.5

En Elasticsearch 5.5, OpenSearch Service admite las siguientes operaciones.

- Todas las operaciones en la ruta de índice (como */index-name* */_forcemerge* y */index-name* */update/id*) excepto */index-name* */_close*
- /_alias
- /_aliases
- /_all
- /_analyze
- /_bulk
- /_cat (excepto /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings para varias propiedades[4]:
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
- /_cluster/state
- /_cluster/stats
- /_count
- /_delete_by_query [1]
- /_explain
- /_field_caps
- /_field_stats
- /_flush
- /_ingest/pipeline
- /_mapping
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_percolate
- /_plugin/kibana
- /_refresh
- /_reindex [1]
- /_render
- /_rollover
- /_scripts [3]
- /_search[2]
- /_search profile
- /_shard_stores
- /_shrink[5]
- /_snapshot
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

1. Los cambios en la configuración de un clúster pueden interrumpir estas operaciones antes de que finalicen. Recomendamos utilizar la operación /_tasks combinada con estas operaciones para verificar que las solicitudes se hayan llevado a cabo correctamente.

2. Las solicitudes DELETE para `/_search/scroll` con un cuerpo de mensaje deben especificar `"Content-Length"` en el encabezado HTTP. La mayoría de los clientes agregan este encabezado de forma predeterminada. Para evitar problemas con caracteres = en valores `scroll_id`, utilice el cuerpo de la solicitud, no la cadena de solicitud, para pasar valores de `scroll_id` a OpenSearch Service.
3. Consulte para ver consideraciones sobre la utilización de scripts the section called “Otros recursos admitidos”.
4. Se refiere al método `PUT`. Para obtener información acerca del método `GET`, consulte the section called “Diferencias de API destacadas”. Esta lista solo hace referencia a las operaciones genéricas de Elasticsearch que OpenSearch Service admite y no incluye operaciones compatibles con complementos específicos para detección de anomalías, ISM, etc.
5. Consulte the section called “Reducir”.

Elasticsearch versión 5.3

En Elasticsearch 5.3, OpenSearch Service admite las siguientes operaciones.

- Todas las operaciones en la ruta de índice (como */index-name* `/_forcemerge` y */index-name* `/update/`*id*) excepto */index-name* `/_close`
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (excepto `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_percolate`
- `/_plugin/kibana`
- `/_render`
- `/_rollover`
- `/_search` [2]
- `/_search profile`
- `/_shard_stores`
- `/_shrink` [4]
- `/_snapshot`
- `/_stats`
- `/_status`
- `/_tasks`
- `/_template`
- `/_update_by_query` [1]
- `/_validate`

- `/_cluster/settings` para varias propiedades[3]:
  - `action.auto_create_index`
  - `action.search.shard_count.limit`
  - `indices.breaker.fielddata.limit`
  - `indices.breaker.request.limit`
  - `indices.breaker.total.limit`
- `/_refresh`
- `/_reindex` [1]

1. Los cambios en la configuración de un clúster pueden interrumpir estas operaciones antes de que finalicen. Recomendamos utilizar la operación `/_tasks` combinada con estas operaciones para verificar que las solicitudes se hayan llevado a cabo correctamente.
2. Las solicitudes DELETE para `/_search/scroll` con un cuerpo de mensaje deben especificar `"Content-Length"` en el encabezado HTTP. La mayoría de los clientes agregan este encabezado de forma predeterminada. Para evitar problemas con caracteres = en valores `scroll_id`, utilice el cuerpo de la solicitud, no la cadena de solicitud, para pasar valores de `scroll_id` a OpenSearch Service.
3. Se refiere al método `PUT`. Para obtener información acerca del método `GET`, consulte the section called “Diferencias de API destacadas”. Esta lista solo hace referencia a las operaciones genéricas de Elasticsearch que OpenSearch Service admite y no incluye operaciones compatibles con complementos específicos para detección de anomalías, ISM, etc.
4. Consulte the section called “Reducir”.

Elasticsearch versión 5.1

En Elasticsearch 5.1, OpenSearch Service admite las siguientes operaciones.

- Todas las operaciones en la ruta de índice (como */index-name* `/_forcemerge` y */index-*
- `/_cluster/state`
- `/_cluster/stats`
- `/_render`
- `/_rollover`

name /update/*id*) excepto /*index-name* /_close
- /_alias
- /_aliases
- /_all
- /_analyze
- /_bulk
- /_cat (excepto /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings para varias propiedades (solo PUT):
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
- /_count
- /_delete_by_query [1]
- /_explain
- /_field_caps
- /_field_stats
- /_flush
- /_ingest/pipeline
- /_mapping
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_percolate
- /_plugin/kibana
- /_refresh
- /_reindex [1]
- /_search[2]
- /_search profile
- /_shard_stores
- /_shrink[3]
- /_snapshot
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

1. Los cambios en la configuración de un clúster pueden interrumpir estas operaciones antes de que finalicen. Recomendamos utilizar la operación /_tasks combinada con estas operaciones para verificar que las solicitudes se hayan llevado a cabo correctamente.
2. Las solicitudes DELETE para /_search/scroll con un cuerpo de mensaje deben especificar "Content-Length" en el encabezado HTTP. La mayoría de los clientes agregan este encabezado de forma predeterminada. Para evitar problemas con caracteres = en valores

`scroll_id`, utilice el cuerpo de la solicitud, no la cadena de solicitud, para pasar valores de `scroll_id` a OpenSearch Service.

3. Consulte the section called “Reducir”.

Elasticsearch versión 2.3

En Elasticsearch 2.3, OpenSearch Service admite las siguientes operaciones.

- Todas las operaciones en la ruta de índice (como `/index-name /_forcemerge` y `/index-name /_recovery`) excepto `/index-name /_close`
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cache/clear` (solo índice)
- `/_cat` (excepto `/_cat/nodeattrs`)
- `/_cluster/health`
- `/_cluster/settings` para varias propiedades (solo PUT):
  - `indices.breaker.fielddata.limit`
  - `indices.breaker.request.limit`
  - `indices.breaker.total.limit`
  - `threadpool.get.queue_size`
  - `threadpool.bulk.queue_size`
  - `threadpool.index.queue_size`
  - `threadpool.percolate.queue_size`
  - `threadpool.search.queue_size`
- `/_cluster/stats`
- `/_count`
- `/_flush`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_nodes`
- `/_percolate`
- `/_plugin/kibana`
- `/_refresh`
- `/_render`
- `/_search`
- `/_snapshot`
- `/_stats`
- `/_status`
- `/_template`

- `threadpool.suggest.queue_size`

Elasticsearch versión 1.5

En Elasticsearch 1.5, OpenSearch Service admite las siguientes operaciones.

- Todas las operaciones de la ruta de índice, como `/`*`index-name`*` /_optimize` y `/`*`index-name`*` /_warmer`, excepto `/`*`index-name`*` /_close`
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat`
- `/_cluster/health`
- `/_cluster/settings` para varias propiedades (solo PUT):
  - `indices.breaker.fielddata.limit`
  - `indices.breaker.request.limit`
  - `indices.breaker.total.limit`
  - `threadpool.get.queue_size`
  - `threadpool.bulk.queue_size`
  - `threadpool.index.queue_size`
  - `threadpool.percolate.queue_size`
  - `threadpool.search.queue_size`
  - `threadpool.suggest.queue_size`
- `/_cluster/stats`
- `/_count`
- `/_flush`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_nodes`
- `/_percolate`
- `/_plugin/kibana`
- `/_plugin/kibana3`
- `/_plugin/migration`
- `/_refresh`
- `/_search`
- `/_snapshot`
- `/_stats`
- `/_status`
- `/_template`

Amazon OpenSearch Service quotas

La cuenta de AWS tiene cuotas predeterminadas para cada servicio de AWS (estas cuotas anteriormente se denominaban “límites”). A menos que se indique lo contrario, cada cuota es específica de la región.

Para ver las cuotas de los dominios e instancias de OpenSearch Service, Amazon OpenSearch sin servidor y Amazon OpenSearch Ingestion, consulte las cuotas de Amazon OpenSearch Service en Referencia general de AWS.

Para ver las cuotas de OpenSearch Service en el Consola de administración de AWS, abra la consola de Service Quotas. En el panel de navegación, seleccione AWSServicios y, a continuación, seleccione Amazon OpenSearch Service. Para solicitar un aumento de cuota, consulte Solicitud de un aumento de cuota en la Guía de usuario de Service Quotas.

Cuotas de almacenamiento UltraWarm

En la tabla siguiente se enumeran los tipos de instancias UltraWarm y la cantidad máxima de almacenamiento que puede utilizar cada tipo. Para obtener más información acerca de UltraWarm, consulte the section called “Almacenamiento UltraWarm”.

Tipo de instancia	Almacenamiento máximo
ultrawarm1.medium.search	1,5 TiB
ultrawarm1.large.search	20 TiB

Número de nodos de datos por AZ

En la siguiente tabla, se indica el número total de nodos de datos para la implementación en una zona de disponibilidad. El límite total indica el número de nodos de datos por límite, incluido el recuento de nodos calientes y calientes, el almacenamiento que puede utilizar cada tipo.

Configuración de AZ	Límite de nodos activos	Límite de nodos cálidos	Límite general (activos + cálido)
1: AZ	334	250	334

Configuración de AZ	Límite de nodos activos	Límite de nodos cálidos	Límite general (activos + cálido)
2: AZ	668	500	668
3: AZ	1002	750	1002

Límite total de nodos por familia de instancias

En la siguiente tabla, se enumera el límite total de nodos por familia de instancias.

Familia de instancias	ElasticSearch OpenSearch hasta la versión 2.15	OpenSearch 2.17 y versiones posteriores	Límite predeterminado
T2	10	10	10
T3	10	10	10
M3, C4, M4, R4, C5, M5, R5, I2, I3	10	200	80
Graviton 2, Gravtion 3, Gravtion 4	200	400	80
C7i, R7i, M7i, i4i, i4g, i8g	200	400	80
OR1.medium.search OR1.large.search OR2.medium.search OR2.large.search OM2.large.search	200	400	80
OR1.xlarge.search y versiones posteriores	200	1002	80

Familia de instancias	ElasticSearch OpenSearch hasta la versión 2.15	OpenSearch 2.17 y versiones posteriores	Límite predeterminado
OR2.xlarge.search y versiones posteriores OM2.xlarge.search y versiones posteriores			
Ultrawarm1	150	750	150

Cuotas de tamaño del volumen de EBS

La siguiente tabla muestra los tamaños mínimo y máximo de los volúmenes de EBS para cada tipo de instancia que OpenSearch Service admite. Para obtener información acerca de qué tipos de instancias incluyen el almacenamiento de instancias y más información sobre el hardware, consulte Precio de Amazon OpenSearch Service.

- Si elige el almacenamiento magnético en Tipo de volumen de EBS al crear un dominio, el tamaño de volumen máximo es de 100 GiB para todos los tipos de instancias, excepto t2.small y t2.medium y todas las instancias de Graviton (M6g, C6g, R6g y R6gd) que no admiten almacenamiento magnético. Para el tamaño máximo que se muestra en la tabla siguiente, elija una de las opciones de SSD.
- Algunos tipos de instancias de generaciones anteriores incluyen almacenamiento de la instancia, pero también admiten el almacenamiento de EBS. Si elige el almacenamiento de EBS para uno de estos tipos de instancia, los volúmenes de almacenamiento no se suman. Puede utilizar el almacenamiento de la instancia o un volumen de EBS, no ambos.

Tipo de instancia	Tamaño mínimo de EBS	Tamaño máximo de EBS (gp2)	Tamaño máximo de EBS (gp3)
t2.micro.search	10 GiB	35 GiB	N/A

Tipo de instancia	Tamaño mínimo de EBS	Tamaño máximo de EBS (gp2)	Tamaño máximo de EBS (gp3)
t2.small.search	10 GiB	35 GiB	N/A
t2.medium.search	10 GiB	35 GiB	N/A
t3.small.search	10 GiB	100 GiB	100 GiB
t3.medium.search	10 GiB	200 GiB	200 GiB
m3.medium.search	10 GiB	100 GiB	N/A
m3.large.search	10 GiB	512 GiB	N/A
m3.xlarge.search	10 GiB	512 GiB	N/A
m3.2xlarge.search	10 GiB	512 GiB	N/A
m4.large.search	10 GiB	512 GiB	N/A
m4.xlarge.search	10 GiB	1 TiB	N/A
m4.2xlarge.search	10 GiB	1,5 TiB	N/A
m4.4xlarge.search	10 GiB	1,5 TiB	N/A
m4.10xlarge.search	10 GiB	1,5 TiB	N/A
m5.large.search	10 GiB	512 GiB	1 TiB
m5.xlarge.search	10 GiB	1 TiB	2 TiB
m5.2xlarge.search	10 GiB	1,5 TiB	3 TiB
m5.4xlarge.search	10 GiB	3 TiB	6 TiB
m5.12xlarge.search	10 GiB	9 TiB	18 TiB
m6g.large.search	10 GiB	512 GiB	1 TiB

Tipo de instancia	Tamaño mínimo de EBS	Tamaño máximo de EBS (gp2)	Tamaño máximo de EBS (gp3)
m6g.xlarge.search	10 GiB	1 TiB	2 TiB
m6g.2xlarge.search	10 GiB	1,5 TiB	3 TiB
m6g.4xlarge.search	10 GiB	3 TiB	6 TiB
m6g.8xlarge.search	10 GiB	6 TiB	12 TiB
m6g.12xlarge.search	10 GiB	9 TiB	18 TiB
c4.large.search	10 GiB	100 GiB	N/A
c4.xlarge.search	10 GiB	512 GiB	N/A
c4.2xlarge.search	10 GiB	1 TiB	N/A
c4.4xlarge.search	10 GiB	1,5 TiB	N/A
c4.8xlarge.search	10 GiB	1,5 TiB	N/A
c5.large.search	10 GiB	256 GiB	256 GiB
c5.xlarge.search	10 GiB	512 GiB	512 GiB
c5.2xlarge.search	10 GiB	1 TiB	1 TiB
c5.4xlarge.search	10 GiB	1,5 TiB	1,5 TiB
c5.9xlarge.search	10 GiB	3,5 TiB	3,5 TiB
c5.18xlarge.search	10 GiB	7 TiB	7 TiB
c6g.large.search	10 GiB	256 GiB	256 GiB
c6g.xlarge.search	10 GiB	512 GiB	512 GiB
c6g.2xlarge.search	10 GiB	1 TiB	1 TiB

Tipo de instancia	Tamaño mínimo de EBS	Tamaño máximo de EBS (gp2)	Tamaño máximo de EBS (gp3)
c6g.4xlarge.search	10 GiB	1,5 TiB	1,5 TiB
c6g.8xlarge.search	10 GiB	3 TiB	3 TiB
c6g.12xlarge.search	10 GiB	4,5 TiB	4,5 TiB
r3.large.search	10 GiB	512 GiB	N/A
r3.xlarge.search	10 GiB	512 GiB	N/A
r3.2xlarge.search	10 GiB	512 GiB	N/A
r3.4xlarge.search	10 GiB	512 GiB	N/A
r3.8xlarge.search	10 GiB	512 GiB	N/A
r4.large.search	10 GiB	1 TiB	N/A
r4.xlarge.search	10 GiB	1,5 TiB	N/A
r4.2xlarge.search	10 GiB	1,5 TiB	N/A
r4.4xlarge.search	10 GiB	1,5 TiB	N/A
r4.8xlarge.search	10 GiB	1,5 TiB	N/A
r4.16xlarge.search	10 GiB	1,5 TiB	N/A
r5.large.search	10 GiB	1 TiB	2 TiB
r5.xlarge.search	10 GiB	1,5 TiB	3 TiB
r5.2xlarge.search	10 GiB	3 TiB	6 TiB
r5.4xlarge.search	10 GiB	6 TiB	12 TiB
r5.12xlarge.search	10 GiB	12 TiB	24 TiB

Tipo de instancia	Tamaño mínimo de EBS	Tamaño máximo de EBS (gp2)	Tamaño máximo de EBS (gp3)
r6g.large.search	10 GiB	1 TiB	2 TiB
r6g.xlarge.search	10 GiB	1,5 TiB	3 TiB
r6g.2xlarge.search	10 GiB	3 TiB	6 TiB
r6g.4xlarge.search	10 GiB	6 TiB	12 TiB
r6g.8xlarge.search	10 GiB	8 TiB	16 TiB
r6g.12xlarge.search	10 GiB	12 TiB	24 TiB
r6gd.large.search	N/A	N/A	N/A
r6gd.xlarge.search	N/A	N/A	N/A
r6gd.2xlarge.search	N/A	N/A	N/A
r6gd.4xlarge.search	N/A	N/A	N/A
r6gd.8xlarge.search	N/A	N/A	N/A
r6gd.12xlarge.search	N/A	N/A	N/A
r6gd.16xlarge.search	N/A	N/A	N/A
i2.xlarge.search	10 GiB	512 GiB	N/A
i2.2xlarge.search	10 GiB	512 GiB	N/A
i3.large.search	N/A	N/A	N/A
i3.xlarge.search	N/A	N/A	N/A
i3.2xlarge.search	N/A	N/A	N/A
i3.4xlarge.search	N/A	N/A	N/A

Tipo de instancia	Tamaño mínimo de EBS	Tamaño máximo de EBS (gp2)	Tamaño máximo de EBS (gp3)
i3.8xlarge.search	N/A	N/A	N/A
i3.16xlarge.search	N/A	N/A	N/A
or1.medium.search	20 GiB	N/A	768 GiB
or1.large.search	20 GiB	N/A	1532 GiB
or1.xlarge.search	20 GiB	N/A	3 TiB
or1.2xlarge.search	20 GiB	N/A	6 TiB
or1.4xlarge.search	20 GiB	N/A	12 TiB
or1.8xlarge.search	20 GiB	N/A	16 TiB
or1.12xlarge.search	20 GiB	N/A	24 TiB
or1.16xlarge.search	20 GiB	N/A	36 TiB
or2.medium.search	20 GiB	N/A	768 GiB
or2.large.search	20 GiB	N/A	1532 GiB
or2.xlarge.search	20 GiB	N/A	3 TiB
or2.2xlarge.search	20 GiB	N/A	6 TiB
or2.4xlarge.search	20 GiB	N/A	12 TiB
or2.8xlarge.search	20 GiB	N/A	16 TiB
or2.12xlarge.search	20 GiB	N/A	24 TiB
or2.16xlarge.search	20 GiB	N/A	36 TiB
om2.large.search	20 GiB	N/A	768 GiB

Tipo de instancia	Tamaño mínimo de EBS	Tamaño máximo de EBS (gp2)	Tamaño máximo de EBS (gp3)
om2.xlarge.search	20 GiB	N/A	2 TiB
om2.2xlarge.search	20 GiB	N/A	3 TiB
om2.4xlarge.search	20 GiB	N/A	6 TiB
om2.8xlarge.search	20 GiB	N/A	12 TiB
om2.12xlarge.search	20 GiB	N/A	18 TiB
om2.16xlarge.search	20 GiB	N/A	24 TiB
im4gn.large.search	N/A	N/A	N/A
im4gn.xlarge.search	N/A	N/A	N/A
im4gn.2xlarge.search	N/A	N/A	N/A
im4gn.4xlarge.search	N/A	N/A	N/A
im4gn.8xlarge.search	N/A	N/A	N/A
im4gn.16xlarge.search	N/A	N/A	N/A
C7g.large.search	10 GiB	N/A	256 GiB
C7g.xlarge.search	10 GiB	N/A	512 GiB
C7g.2xlarge.search	10 GiB	N/A	1 TiB
C7g.4xlarge.search	10 GiB	N/A	1,5 TiB
C7g.8xlarge.search	10 GiB	N/A	3 TiB
C7g.12xlarge.search	10 GiB	N/A	4,5 TiB
C7g.16xlarge.search	10 GiB	N/A	6 TiB

Tipo de instancia	Tamaño mínimo de EBS	Tamaño máximo de EBS (gp2)	Tamaño máximo de EBS (gp3)
M7g.medium.search	10 GiB	N/A	512 GiB
M7g.large.search	10 GiB	N/A	768 GiB
M7g.xlarge.search	10 GiB	N/A	2 TiB
M7g.2xlarge.search	10 GiB	N/A	3 TiB
M7g.4xlarge.search	10 GiB	N/A	6 TiB
M7g.8xlarge.search	10 GiB	N/A	12 TiB
M7g.12xlarge.search	10 GiB	N/A	18 TiB
M7g.16xlarge.search	10 GiB	N/A	24 TiB
R7g.medium.search	10 GiB	N/A	768 GiB
R7g.large.search	10 GiB	N/A	1,5 TiB
R7g.xlarge.search	10 GiB	N/A	3 TiB
R7g.2xlarge.search	10 GiB	N/A	6 TiB
R7g.4xlarge.search	10 GiB	N/A	12 TiB
R7g.8xlarge.search	10 GiB	N/A	16 TiB
R7g.12xlarge.search	10 GiB	N/A	24 TiB
R7g.16xlarge.search	10 GiB	N/A	36 TiB
R7gd.large.search	N/A	N/A	N/A
R7gd.xlarge.search	N/A	N/A	N/A
R7gd.2xlarge.search	N/A	N/A	N/A

Tipo de instancia	Tamaño mínimo de EBS	Tamaño máximo de EBS (gp2)	Tamaño máximo de EBS (gp3)
R7gd.4xlarge.search	N/A	N/A	N/A
R7gd.8xlarge.search	N/A	N/A	N/A
R7gd.12xlarge.search	N/A	N/A	N/A
R7gd.16xlarge.search	N/A	N/A	N/A
i4i.large.search	N/A	N/A	N/A
i4i.xlarge.search	N/A	N/A	N/A
i4i.2xlarge.search	N/A	N/A	N/A
i4i.4xlarge.search	N/A	N/A	N/A
i4i.8xlarge.search	N/A	N/A	N/A
i4i.12xlarge.search	N/A	N/A	N/A
i4i.16xlarge.search	N/A	N/A	N/A
i4i.24xlarge.search	N/A	N/A	N/A
i4i.32xlarge.search	N/A	N/A	N/A
i4g.large.search	N/A	N/A	N/A
i4g.xlarge.search	N/A	N/A	N/A
i4g.2xlarge.search	N/A	N/A	N/A
i4g.4xlarge.search	N/A	N/A	N/A
i4g.8xlarge.search	N/A	N/A	N/A
i4g.16xlarge.search	N/A	N/A	N/A

Tipo de instancia	Tamaño mínimo de EBS	Tamaño máximo de EBS (gp2)	Tamaño máximo de EBS (gp3)
i8g.large.search	N/A	N/A	N/A
i8g.xlarge.search	N/A	N/A	N/A
i8g.2xlarge.search	N/A	N/A	N/A
i8g.4xlarge.search	N/A	N/A	N/A
i8g.8xlarge.search	N/A	N/A	N/A
i8g.12xlarge.search	N/A	N/A	N/A
i8g.16xlarge.search	N/A	N/A	N/A
c7i.large.search	10 GiB	N/A	256 GiB
c7i.xlarge.search	10 GiB	N/A	512 GiB
c7i.2xlarge.search	10 GiB	N/A	1 TiB
c7i.4xlarge.search	10 GiB	N/A	1,5 TiB
c7i.8xlarge.search	10 GiB	N/A	3 TiB
c7i.12xlarge.search	10 GiB	N/A	4,5 TiB
c7i.16xlarge.search	10 GiB	N/A	6 TiB
m7i.large.search	10 GiB	N/A	768 GiB
m7i.xlarge.search	10 GiB	N/A	2 TiB
m7i.2xlarge.search	10 GiB	N/A	3 TiB
m7i.4xlarge.search	10 GiB	N/A	6 TiB
m7i.8xlarge.search	10 GiB	N/A	12 TiB

Tipo de instancia	Tamaño mínimo de EBS	Tamaño máximo de EBS (gp2)	Tamaño máximo de EBS (gp3)
m7i.12xlarge.search	10 GiB	N/A	18 TiB
m7i.16xlarge.search	10 GiB	N/A	24 TiB
r7i.large.search	10 GiB	N/A	1,5 TiB
r7i.xlarge.search	10 GiB	N/A	3 TiB
r7i.2xlarge.search	10 GiB	N/A	6 TiB
r7i.4xlarge.search	10 GiB	N/A	12 TiB
r7i.8xlarge.search	10 GiB	N/A	16 TiB
r7i.12xlarge.search	10 GiB	N/A	24 TiB
r7i.12xlarge.search	10 GiB	N/A	36 TiB
c8g.large	10 GiB	N/A	256 GiB
c8g.xlarge	10 GiB	N/A	512 GiB
c8g.2xlarge	10 GiB	N/A	1 TiB
c8g.4xlarge	10 GiB	N/A	1,5 TiB
c8g.8xlarge	10 GiB	N/A	3 TiB
c8g.12xlarge	10 GiB	N/A	4,5 TiB
c8g.16xlarge	10 GiB	N/A	6 TiB
m8g.medium	10 GiB	N/A	512 GiB
m8g.large	10 GiB	N/A	768 GiB
m8g.xlarge	10 GiB	N/A	2 TiB

Tipo de instancia	Tamaño mínimo de EBS	Tamaño máximo de EBS (gp2)	Tamaño máximo de EBS (gp3)
m8g.2xlarge	10 GiB	N/A	3 TiB
m8g.4xlarge	10 GiB	N/A	6 TiB
m8g.8xlarge	10 GiB	N/A	12 TiB
m8g.12xlarge	10 GiB	N/A	18 TiB
m8g.16xlarge	10 GiB	N/A	24 TiB
r8g.medium	10 GiB	N/A	768 GiB
r8g.large	10 GiB	N/A	1532 GiB
r8g.xlarge	10 GiB	N/A	3 TiB
r8g.2xlarge	10 GiB	N/A	6 TiB
r8g.4xlarge	10 GiB	N/A	12 TiB
r8g.8xlarge	10 GiB	N/A	16 TiB
r8g.12xlarge	10 GiB	N/A	24 TiB
r8g.16xlarge	10 GiB	N/A	36 TiB
r8gd.medium	N/A	N/A	N/A
r8gd.large	N/A	N/A	N/A
r8gd.xlarge	N/A	N/A	N/A
r8gd.2xlarge	N/A	N/A	N/A
r8gd.4xlarge	N/A	N/A	N/A
r8gd.8xlarge	N/A	N/A	N/A

Tipo de instancia	Tamaño mínimo de EBS	Tamaño máximo de EBS (gp2)	Tamaño máximo de EBS (gp3)
r8gd.12xlarge	N/A	N/A	N/A
r8gd.16xlarge	N/A	N/A	N/A

Cuotas de red

La siguiente tabla muestra el tamaño máximo de las cargas de solicitudes HTTP.

Tipo de instancia	Tamaño máximo de las cargas de solicitud HTTP
t2.micro.search	10 MiB
t2.small.search	10 MiB
t2.medium.search	10 MiB
t3.small.search	10 MiB
t3.medium.search	10 MiB
m3.medium.search	10 MiB
m3.large.search	10 MiB
m3.xlarge.search	100 MiB
m3.2xlarge.search	100 MiB
m4.large.search	10 MiB
m4.xlarge.search	100 MiB
m4.2xlarge.search	100 MiB
m4.4xlarge.search	100 MiB

Tipo de instancia	Tamaño máximo de las cargas de solicitud HTTP
m4.10xlarge.search	100 MiB
m5.large.search	10 MiB
m5.xlarge.search	100 MiB
m5.2xlarge.search	100 MiB
m5.4xlarge.search	100 MiB
m5.12xlarge.search	100 MiB
m6g.large.search	10 MiB
m6g.xlarge.search	100 MiB
m6g.2xlarge.search	100 MiB
m6g.4xlarge.search	100 MiB
m6g.8xlarge.search	100 MiB
m6g.12xlarge.search	100 MiB
c4.large.search	10 MiB
c4.xlarge.search	100 MiB
c4.2xlarge.search	100 MiB
c4.4xlarge.search	100 MiB
c4.8xlarge.search	100 MiB
c5.large.search	10 MiB
c5.xlarge.search	100 MiB
c5.2xlarge.search	100 MiB

Tipo de instancia	Tamaño máximo de las cargas de solicitud HTTP
c5.4xlarge.search	100 MiB
c5.9xlarge.search	100 MiB
c5.18xlarge.search	100 MiB
c6g.large.search	10 MiB
c6g.xlarge.search	100 MiB
c6g.2xlarge.search	100 MiB
c6g.4xlarge.search	100 MiB
c6g.8xlarge.search	100 MiB
c6g.12xlarge.search	100 MiB
r3.large.search	10 MiB
r3.xlarge.search	100 MiB
r3.2xlarge.search	100 MiB
r3.4xlarge.search	100 MiB
r3.8xlarge.search	100 MiB
r4.large.search	100 MiB
r4.xlarge.search	100 MiB
r4.2xlarge.search	100 MiB
r4.4xlarge.search	100 MiB
r4.8xlarge.search	100 MiB
r4.16xlarge.search	100 MiB

Tipo de instancia	Tamaño máximo de las cargas de solicitud HTTP
r5.large.search	100 MiB
r5.xlarge.search	100 MiB
r5.2xlarge.search	100 MiB
r5.4xlarge.search	100 MiB
r5.12xlarge.search	100 MiB
r6g.large.search	100 MiB
r6g.xlarge.search	100 MiB
r6g.2xlarge.search	100 MiB
r6g.4xlarge.search	100 MiB
r6g.8xlarge.search	100 MiB
r6g.12xlarge.search	100 MiB
r6gd.large.search	100 MiB
r6gd.xlarge.search	100 MiB
r6gd.2xlarge.search	100 MiB
r6gd.4xlarge.search	100 MiB
r6gd.8xlarge.search	100 MiB
r6gd.12xlarge.search	100 MiB
r6gd.16xlarge.search	100 MiB
i2.xlarge.search	100 MiB
i2.2xlarge.search	100 MiB

Tipo de instancia	Tamaño máximo de las cargas de solicitud HTTP
i3.large.search	100 MiB
i3.xlarge.search	100 MiB
i3.2xlarge.search	100 MiB
i3.4xlarge.search	100 MiB
i3.8xlarge.search	100 MiB
i3.16xlarge.search	100 MiB
or1.medium.search	10 MiB
or1.large.search	100 MiB
or1.xlarge.search	100 MiB
or1.2xlarge.search	100 MiB
or1.4xlarge.search	100 MiB
or1.8xlarge.search	100 MiB
or1.12xlarge.search	100 MiB
or1.16xlarge.search	100 MiB
or2.medium.search	100 MiB
or2.large.search	100 MiB
or2.xlarge.search	100 MiB
or2.2xlarge.search	100 MiB
or2.4xlarge.search	100 MiB
or2.8xlarge.search	100 MiB

Tipo de instancia	Tamaño máximo de las cargas de solicitud HTTP
or2.12xlarge.search	100 MiB
or2.16xlarge.search	100 MiB
om2.large.search	10 MiB
om2.xlarge.search	100 MiB
om2.2xlarge.search	100 MiB
om2.4xlarge.search	100 MiB
om2.8xlarge.search	100 MiB
om2.12xlarge.search	100 MiB
om2.16xlarge.search	100 MiB
im4gn.large.search	100 MiB
im4gn.xlarge.search	100 MiB
im4gn.2xlarge.search	100 MiB
im4gn.4xlarge.search	100 MiB
im4gn.8xlarge.search	100 MiB
im4gn.16xlarge.search	100 MiB
i4i.large.search	100 MiB
i4i.xlarge.search	100 MiB
i4i.2xlarge.search	100 MiB
i4i.4xlarge.search	100 MiB
i4i.8xlarge.search	100 MiB

Tipo de instancia	Tamaño máximo de las cargas de solicitud HTTP
i4i.12xlarge.search	100 MiB
i4i.16xlarge.search	100 MiB
i4i.24xlarge.search	100 MiB
i4i.32xlarge.search	100 MiB
i4g.large.search	100 MiB
i4g.xlarge.search	100 MiB
i4g.2xlarge.search	100 MiB
i4g.4xlarge.search	100 MiB
i4g.8xlarge.search	100 MiB
i4g.16xlarge.search	100 MiB
i8g.large.search	100 MiB
i8g.xlarge.search	100 MiB
i8g.2xlarge.search	100 MiB
i8g.4xlarge.search	100 MiB
i8g.8xlarge.search	100 MiB
i8g.12xlarge.search	100 MiB
i8g.16xlarge.search	100 MiB
c7i.large.search	100 MiB
c7i.xlarge.search	100 MiB
c7i.2xlarge.search	100 MiB

Tipo de instancia	Tamaño máximo de las cargas de solicitud HTTP
c7i.4xlarge.search	100 MiB
c7i.8xlarge.search	100 MiB
c7i.12xlarge.search	100 MiB
c7i.16xlarge.search	100 MiB
m7i.large.search	100 MiB
m7i.xlarge.search	100 MiB
m7i.2xlarge.search	100 MiB
m7i.4xlarge.search	100 MiB
m7i.8xlarge.search	100 MiB
m7i.12xlarge.search	100 MiB
m7i.16xlarge.search	100 MiB
r7i.large.search	100 MiB
r7i.xlarge.search	100 MiB
r7i.2xlarge.search	100 MiB
r7i.4xlarge.search	100 MiB
r7i.8xlarge.search	100 MiB
r7i.12xlarge.search	100 MiB
r7i.16xlarge.search	100 MiB
c8g.large	100 MiB
c8g.xlarge	100 MiB

Tipo de instancia	Tamaño máximo de las cargas de solicitud HTTP
c8g.2xlarge	100 MiB
c8g.4xlarge	100 MiB
c8g.8xlarge	100 MiB
c8g.12xlarge	100 MiB
c8g.16xlarge	100 MiB
m8g.medium	100 MiB
m8g.large	100 MiB
m8g.xlarge	100 MiB
m8g.2xlarge	100 MiB
m8g.4xlarge	100 MiB
m8g.8xlarge	100 MiB
m8g.12xlarge	100 MiB
m8g.16xlarge	100 MiB
r8g.medium	100 MiB
r8g.large	100 MiB
r8g.xlarge	100 MiB
r8g.2xlarge	100 MiB
r8g.4xlarge	100 MiB
r8g.8xlarge	100 MiB
r8g.12xlarge	100 MiB

Tipo de instancia	Tamaño máximo de las cargas de solicitud HTTP
r8g.16xlarge	100 MiB
r8gd.medium	100 MiB
r8gd.large	100 MiB
r8gd.xlarge	100 MiB
r8gd.2xlarge	100 MiB
r8gd.4xlarge	100 MiB
r8gd.8xlarge	100 MiB
r8gd.12xlarge	100 MiB
r8gd.16xlarge	100 MiB

Cuotas de tamaño de la partición

En la siguiente sección, se enumeran los tamaños máximos de las particiones para varias familias de instancias.

Tipo de instancia	Multi-AZ sin modo de espera	Multi-AZ con modo de espera
R5, C5, M5, C7i, M7i, R7i	N/A	65 GiB
I3, i4i, i4g, i8g	N/A	65 GiB
R6g, C6g, M6g, R6gd, C7g,M7g, R7g, R7gd, C8g, M8g, R8g, R8gd	N/A	65 GiB
OR1, OR2, OM2	100 GiB	65 GiB
Im4gn	N/A	65 GiB

Para solicitar un aumento de cuota, póngase en contacto con AWS Support.

Cuotas de recuento de la partición

La siguiente sección enumera el número máximo de particiones para las versiones de OpenSearch.

Versión del motor	Límite	Notas
Elasticsearch de la versión 1.5 a la 6.x	Sin valor predeterminado	
Elasticsearch 7.x	1 000	El límite predeterminado se puede cambiar mediante la configuración de clúster. max_shards_per_node.
OpenSearch de la versión 1.x a la 2.15	1 000	El límite predeterminado se puede cambiar mediante la configuración de clúster. max_shards_per_node.
OpenSearch 2.17 y versiones posteriores	1000 por cada 16 GB de heap hasta un máximo de 4000	El límite predeterminados no se puede cambiar.

Cuota de procesamiento de Java

OpenSearch Service limita los procesos de Java a un tamaño de pila de 32 GiB. Los usuarios avanzados pueden especificar el porcentaje del montón usado para los datos de campo. Para obtener más información, consulte the section called “Configuración avanzada de clústeres” y the section called “OutOfMemoryError de JVM”.

Cuota de políticas de dominio

OpenSearch Service limita las políticas de acceso en dominios a 100 KiB.

Instancias reservadas en Amazon OpenSearch Service

Las instancias reservadas (RI) en Amazon OpenSearch Service ofrecen descuentos importantes en comparación con las instancias bajo demanda. Las instancias en sí son idénticas; las instancias reservadas simplemente suponen que se aplica un descuento de facturación con respecto al uso de instancias bajo demanda en su cuenta. Para las aplicaciones de larga duración con un uso predecible, las instancias reservadas pueden proporcionar un ahorro considerable a lo largo del tiempo.

Las RI de OpenSearch Service requieren condiciones de uno o tres años y tienen tres opciones de pago que afectan a la tarifa de descuento:

- Sin pago inicial: no se paga nada por adelantado. Se paga una tarifa por hora con descuento por cada hora dentro del periodo estipulado.
- Pago inicial parcial: se paga una parte del costo por adelantado y después una tarifa por hora con descuento por cada hora dentro del periodo estipulado.
- Pago total anticipado: se paga la totalidad del costo por adelantado. No hay que pagar una tarifa por hora durante el periodo estipulado.

Por regla general, un pago inicial más grande significa un mayor descuento. Las instancias reservadas no se pueden cancelar: cuando se reservan, se compromete a pagar por el periodo completo, y los pagos iniciales no son reembolsables.

Las RI no son flexibles; solo se aplican al tipo de instancia exacto que se reserva. Por ejemplo, una reserva para ocho instancias `c5.2xlarge.search` no se aplica a dieciséis instancias `c5.xlarge.search` o cuatro instancias `c5.4xlarge.search`. Sin embargo, las cuentas vinculadas que forman parte de una organización en AWS Organizations pueden beneficiarse de cualquier solicitud de descuento no utilizada de la cuenta propietaria de la RI, siempre que los tipos de instancia, la región, la familia y el tamaño coincidan. Para obtener más información, consulte los Precios de Amazon OpenSearch Service y las preguntas frecuentes.

Comprar instancias reservadas (consola)

La consola le permite ver las instancias reservadas que tiene y adquirir otras nuevas.

Para comprar una reserva

1. Visite https://aws.amazon.com y, a continuación, seleccione Iniciar sesión en la consola.

2. En Análisis, seleccione Amazon OpenSearch Service.
3. Seleccione Reserved Instance Leases (Asignaciones de instancias reservadas) en el panel de navegación.

   En esta página, puede ver la reservas que tiene. Si tiene muchas, puede filtrarlas para identificar y ver más fácilmente una reserva en concreto.

   > **Tip**
   >
   > Si no ve el enlace de Reserved Instances Leases (Asignaciones de instancias reservadas), cree un dominio en la Región de AWS.

4. Seleccione Order reserved instance (Pedir instancia reservada).
5. Proporcione un nombre único y descriptivo.
6. Seleccione un tipo de instancia y el número de instancias. Para obtener instrucciones, consulte the section called “Determinación del tamaño de dominios”.
7. Seleccione la duración del periodo y una opción de pago. Revise los datos de pago con cuidado.
8. Seleccione Next (Siguiente).
9. Revise el resumen de compra con cuidado. Las compras de instancias reservadas no son reembolsables.
10. Seleccione Order (Pedir).

Comprar instancias reservadas (CLI de AWS)

La AWS CLI tiene comandos para ver las ofertas, comprar una reserva y consultar las reservas que se tienen. El comando y la respuesta de muestra siguientes muestran las ofertas para una determinada Región de AWS:

```
aws opensearch describe-reserved-instance-offerings --region us-east-1
{
  "ReservedInstanceOfferings": [
    {
      "FixedPrice": x,
      "ReservedInstanceOfferingId": "1a2a3a4a5-1a2a-3a4a-5a6a-1a2a3a4a5a6a",
      "RecurringCharges": [
        {
          "RecurringChargeAmount": y,
```

```
          "RecurringChargeFrequency": "Hourly"
        }
      ],
      "UsagePrice": 0.0,
      "PaymentOption": "PARTIAL_UPFRONT",
      "Duration": 31536000,
      "InstanceType": "m4.2xlarge.search",
      "CurrencyCode": "USD"
    }
  ]
}
```

Para ver una explicación de cada valor de retorno, consulte la tabla siguiente.

Campo	Descripción
`FixedPrice`	El costo inicial de la reserva.
`ReservedInstanceOfferingId`	El ID de la oferta. Anote este valor si desea reservar la oferta.
`RecurringCharges`	La tarifa por hora de la reserva.
`UsagePrice`	Campo heredado. Para OpenSearch Service, este valor siempre es 0.
`PaymentOption`	No Upfront (Sin pago inicial), Partial Upfront (Pago inicial parcial) o All Upfront (Pago inicial total).
`Duration`	La duración en segundos: • 31536000 segundos equivale a un año. • 94608000 segundos equivale a tres años.
`InstanceType`	El tipo de instancia de la reserva. Para obtener información sobre los recursos de equipo que se asignan a cada tipo de instancia, consulte Precios de Amazon OpenSearch Service.

Campo	Descripción
`CurrencyCode`	La divisa para `FixedPrice` y `RecurringChargeAmount`.

El siguiente ejemplo adquiere una reserva:

```
aws opensearch purchase-reserved-instance-offering --reserved-instance-offering-
id 1a2a3a4a5-1a2a-3a4a-5a6a-1a2a3a4a5a6a --reservation-name my-reservation --instance-
count 3 --region us-east-1
{
  "ReservationName": "my-reservation",
  "ReservedInstanceId": "9a8a7a6a-5a4a-3a2a-1a0a-9a8a7a6a5a4a"
}
```

Por último, puede obtener una lista de las reservas para una región determinada mediante el siguiente ejemplo:

```
aws opensearch describe-reserved-instances --region us-east-1
{
  "ReservedInstances": [
    {
      "FixedPrice": x,
      "ReservedInstanceOfferingId": "1a2a3a4a5-1a2a-3a4a-5a6a-1a2a3a4a5a6a",
      "ReservationName": "my-reservation",
      "PaymentOption": "PARTIAL_UPFRONT",
      "UsagePrice": 0.0,
      "ReservedInstanceId": "9a8a7a6a-5a4a-3a2a-1a0a-9a8a7a6a5a4a",
      "RecurringCharges": [
        {
          "RecurringChargeAmount": y,
          "RecurringChargeFrequency": "Hourly"
        }
      ],
      "State": "payment-pending",
      "StartTime": 1522872571.229,
      "InstanceCount": 3,
      "Duration": 31536000,
      "InstanceType": "m4.2xlarge.search",
      "CurrencyCode": "USD"
    }
```

```
  ]
}
```

> **Note**
>
> `StartTime` se expresa en formato de tiempo Unix, que es el número de segundos que han transcurrido desde la medianoche UTC del 1 de enero de 1970. Por ejemplo, 1522872571 en formato de tiempo Unix son las 20:09:31 UTC del 4 de abril de 2018. Puede utilizar convertidores online.

Para obtener más información acerca de los comandos utilizados en los ejemplos anteriores, consulte la Referencia de comandos de la AWS CLI.

Comprar instancias reservadas (SDK de AWS)

Los AWS SDK (excepto los SDK de Android e iOS) admiten todas las operaciones definidas en la Referencia de la API de configuración de Amazon OpenSearch Service, incluidas las siguientes:

- `DescribeReservedInstanceOfferings`
- `PurchaseReservedInstanceOffering`
- `DescribeReservedInstances`

En este ejemplo de script se utiliza el cliente Python de bajo nivel OpenSearchService desde AWS SDK para Python (Boto3) para adquirir instancias reservadas. Se debe proporcionar un valor para `instance_type`.

```
import boto3
from botocore.config import Config

# Build the client using the default credential configuration.
# You can use the CLI and run 'aws configure' to set access key, secret
# key, and default region.

my_config = Config(
    # Optionally lets you specify a region other than your default.
    region_name='us-east-1'
)
```

```
client = boto3.client('opensearch', config=my_config)

instance_type = '' # e.g. m4.2xlarge.search


def describe_RI_offerings(client):
    """Gets the Reserved Instance offerings for this account"""

    response = client.describe_reserved_instance_offerings()
    offerings = (response['ReservedInstanceOfferings'])
    return offerings


def check_instance(offering):
    """Returns True if instance type is the one you specified above"""

    if offering['InstanceType'] == instance_type:
        return True

    return False


def get_instance_id():
    """Iterates through the available offerings to find the ID of the one you
 specified"""

    instance_type_iterator = filter(
        check_instance, describe_RI_offerings(client))
    offering = list(instance_type_iterator)
    id = offering[0]['ReservedInstanceOfferingId']
    return id


def purchase_RI_offering(client):
    """Purchase Reserved Instances"""

    response = client.purchase_reserved_instance_offering(
        ReservedInstanceOfferingId = get_instance_id(),
        ReservationName = 'my-reservation',
        InstanceCount = 1
    )
    print('Purchased reserved instance offering of type ' + instance_type)
    print(response)
```

```
def main():
    """Purchase Reserved Instances"""
    purchase_RI_offering(client)
```

Para obtener más información acerca de cómo instalar y utilizar los SDK de AWS, consulte los Kits de desarrollo de software de AWS.

Examinar los costos

El Explorador de costos es una herramienta gratuita que puede utilizar para ver sus datos de gasto de los últimos 13 meses. El análisis de estos datos lo ayuda a identificar tendencias y a comprender si las instancias reservadas se adaptan a su caso de uso. Si ya tiene RI, puede agrupar por Purchase Option (Opción de compra) y mostrar los costos amortizados para comparar ese gasto con su gasto en instancias bajo demanda. También puede definir presupuestos de utilización para asegurarse de que aprovecha al máximo sus reservas. Para obtener más información, consulte Análisis de los costos con Cost Explorer en la Guía del usuario de AWS Billing.

Otros recursos admitidos en Amazon OpenSearch Service

En este tema se describen los recursos adicionales que admite Amazon OpenSearch Service.

bootstrap.memory_lock

OpenSearch Service habilita `bootstrap.memory_lock` en `opensearch.yml`, que bloquea la memoria de JVM e impide que el sistema operativo la cambie al disco. Esto se aplica a todos los tipos de instancias admitidos excepto a los siguientes:

- `t2.micro.search`
- `t2.small.search`
- `t2.medium.search`
- `t3.small.search`
- `t3.medium.search`

Módulo de scripting

OpenSearch Service admite el scripting para dominios de Elasticsearch 5.x y posteriores. No admite scripting para las versiones 1.5 o 2.3.

Las opciones de scripting admitidas incluyen lo siguiente:

- Painless
- Lucene Expressions
- Mustache

En todos los dominios de OpenSearch, y los dominios de Elasticsearch 5.5 y posteriores, OpenSearch Service admite scripts almacenados mediante el punto de conexión `_scripts`. Los dominios de Elasticsearch 5.3 y 5.1 solo admiten scripts incorporados.

Transporte TLS

OpenSearch Service admite HTTP en el puerto 80 y HTTPS en el puerto 443, pero no el transporte TLS.

Tutoriales de Amazon OpenSearch Service

Este capítulo incluye varios tutoriales completos para trabajar con Amazon OpenSearch Service, incluyendo cómo migrar al servicio, crear una aplicación de búsqueda simple y crear una visualización en OpenSearch Dashboards.

Temas

- Tutorial: Creación y búsqueda de documentos en Amazon OpenSearch Service
- Tutorial: Migración a Amazon OpenSearch Service
- Tutorial: Creación de una aplicación de búsqueda con Amazon OpenSearch Service
- Tutorial: Visualización de llamadas de soporte al cliente con OpenSearch Service y OpenSearch Dashboards

Tutorial: Creación y búsqueda de documentos en Amazon OpenSearch Service

En este tutorial, aprenderá a crear y buscar un documento en Amazon OpenSearch Service. Agregue datos a un índice en forma de documento JSON. OpenSearch Service crea un índice alrededor del primer documento que agrega.

Este tutorial explica cómo llevar a cabo solicitudes HTTP para crear documentos, generar automáticamente un ID para un documento y hacer búsquedas básicas y avanzadas en los documentos.

> **Note**
>
> En este tutorial, se utiliza un dominio de acceso abierto. Para obtener el máximo nivel de seguridad, es recomendable que coloque su dominio dentro de una nube privada virtual (VPC).

Requisitos previos

Este tutorial tiene los requisitos previos siguientes:

- Debe tener una Cuenta de AWS.
- Debe tener un dominio de OpenSearch Service activo.

Adición de un documento a un índice

Para agregar un documento a un índice, puede utilizar cualquier herramienta HTTP, como Postman, cURL o la consola de paneles de OpenSearch. En estos ejemplos, se supone que está utilizando la consola para desarrolladores en los paneles de OpenSearch. Si utiliza una herramienta diferente, ajústela según corresponda proporcionando la URL completa y las credenciales, si es necesario.

Para agregar un documento a un índice

1. Desplácese hasta la URL de OpenSearch Dashboards para su dominio. Puede encontrar la URL en el panel del dominio en la consola de OpenSearch Service. La URL tiene este formato:

   ```
   domain-endpoint/_dashboards/
   ```

2. Inicie sesión con su nombre de usuario y contraseña principales.
3. Abra el panel de navegación izquierdo y elija Herramientas para desarrolladores.
4. El verbo HTTP para crear un nuevo recurso es PUT, que es lo que se utiliza para crear un nuevo documento e índice. Ingrese el siguiente comando en la consola:

   ```
   PUT fruit/_doc/1
   {
     "name":"strawberry",
     "color":"red"
   }
   ```

   La solicitud PUT crea un índice llamado fruit (fruta) y agrega un solo documento al índice con un ID de 1. Produce la siguiente respuesta:

   ```
   {
     "_index" : "fruit",
     "_type" : "_doc",
     "_id" : "1",
     "_version" : 1,
     "result" : "created",
     "_shards" : {
       "total" : 2,
   ```

```
    "successful" : 2,
    "failed" : 0
  },
  "_seq_no" : 0,
  "_primary_term" : 1
}
```

Creación de ID de generación automática

OpenSearch Service puede generar automáticamente un ID para sus documentos. El comando para generar ID utiliza una solicitud POST en lugar de una solicitud PUT y no requiere ningún ID de documento (en comparación con la solicitud anterior).

Ingrese la siguiente solicitud en la consola para desarrolladores:

```
POST veggies/_doc
{
  "name":"beet",
  "color":"red",
  "classification":"root"
}
```

Esta solicitud crea un índice denominado veggies (verduras) y agrega el documento al índice. Produce la siguiente respuesta:

```
{
  "_index" : "veggies",
  "_type" : "_doc",
  "_id" : "3WgyS4IB5DLqbRIvLxtF",
  "_version" : 1,
  "result" : "created",
  "_shards" : {
    "total" : 2,
    "successful" : 2,
    "failed" : 0
  },
  "_seq_no" : 0,
  "_primary_term" : 1
}
```

Tenga en cuenta el campo `_id` adicional de la respuesta, que indica que se creó un ID automáticamente.

> **Note**
>
> No proporcione nada después de `_doc` en la URL, donde normalmente va el ID. Como va a crear un documento con un ID generado, aún no debe proporcionar uno. Eso está reservado para actualizaciones.

Actualización de un documento con un comando POST

Para actualizar un documento, se utiliza un comando `POST` de HTTP con el número de ID.

En primer lugar, cree un documento con un ID de `42`:

```
POST fruits/_doc/42
{
  "name":"banana",
  "color":"yellow"
}
```

A continuación, utilice ese ID para actualizar el documento:

```
POST fruits/_doc/42
{
  "name":"banana",
  "color":"yellow",
  "classification":"berries"
}
```

Este comando actualiza el documento con el nuevo campo `classification`. Produce la siguiente respuesta:

```
{
  "_index" : "fruits",
  "_type" : "_doc",
  "_id" : "42",
  "_version" : 2,
  "result" : "updated",
  "_shards" : {
```

```
    "total" : 2,
    "successful" : 2,
    "failed" : 0
  },
  "_seq_no" : 1,
  "_primary_term" : 1
}
```

> **Note**
>
> Si intenta actualizar un documento que no existe, OpenSearch Service crea el documento.

Ejecución de acciones por lotes

Puede utilizar la operación de la API de `POST _bulk` para llevar a cabo varias acciones en uno o más índices en una solicitud. Los comandos de acción por lotes tienen el siguiente formato:

```
POST /_bulk
<action_meta>\n
<action_data>\n
<action_meta>\n
<action_data>\n
```

Cada acción requiere dos líneas de JSON. Primero, debe proporcionar la descripción de la acción o los metadatos. En la línea siguiente, debe proporcionar los datos. Cada parte está separada por una nueva línea (\n). La descripción de una acción de una inserción podría verse de la siguiente manera:

```
{ "create" : { "_index" : "veggies", "_type" : "_doc", "_id" : "7" } }
```

La línea siguiente que contiene los datos podría verse de la siguiente manera:

```
{ "name":"kale", "color":"green", "classification":"leafy-green" }
```

En conjunto, los metadatos y los datos representan una sola acción en una operación masiva. Puede llevar a cabo muchas operaciones en una sola solicitud, como esta:

```
POST /_bulk
{ "create" : { "_index" : "veggies", "_id" : "35" } }
{ "name":"kale", "color":"green", "classification":"leafy-green" }
```

```
{ "create" : { "_index" : "veggies", "_id" : "36" } }
{ "name":"spinach", "color":"green", "classification":"leafy-green" }
{ "create" : { "_index" : "veggies", "_id" : "37" } }
{ "name":"arugula", "color":"green", "classification":"leafy-green" }
{ "create" : { "_index" : "veggies", "_id" : "38" } }
{ "name":"endive", "color":"green", "classification":"leafy-green" }
{ "create" : { "_index" : "veggies", "_id" : "39" } }
{ "name":"lettuce", "color":"green", "classification":"leafy-green" }
{ "delete" : { "_index" : "vegetables", "_id" : "1" } }
```

Observe que la última acción es `delete`. No hay datos después de la acción `delete`.

Búsqueda de documentos

Ya que existen datos en el clúster, puede buscarlos. Por ejemplo, es posible que quiera buscar todos los tubérculos, obtener un recuento de todos los vegetales de hoja verde o encontrar la cantidad de errores registrados por hora.

Búsquedas básicas

Una búsqueda básica tiene un aspecto similar a este:

```
GET veggies/_search?q=name:l*
```

La solicitud produce una respuesta JSON que contiene el documento de lechuga.

Búsquedas avanzadas

Puede hacer búsquedas más avanzadas si proporciona las opciones de consulta como JSON en el cuerpo de la solicitud:

```
GET veggies/_search
{
  "query": {
    "term": {
      "name": "lettuce"
    }
  }
}
```

Este ejemplo también produce una respuesta JSON con el documento de lechuga.

Ordenar

Puede llevar a cabo más consultas de este tipo mediante la acción de ordenar. En primer lugar, debe volver a crear el índice porque la asignación automática de campos eligió tipos que no se pueden ordenar de forma predeterminada. Envíe las siguientes solicitudes para eliminar y volver a crear el índice:

```
DELETE /veggies

PUT /veggies
{
   "mappings":{
      "properties":{
         "name":{
            "type":"keyword"
         },
         "color":{
            "type":"keyword"
         },
         "classification":{
            "type":"keyword"
         }
      }
   }
}
```

A continuación, rellene el índice con datos:

```
POST /_bulk
{ "create" : { "_index" : "veggies", "_id" : "7"  } }
{ "name":"kale", "color":"green", "classification":"leafy-green" }
{ "create" : { "_index" : "veggies", "_id" : "8" } }
{ "name":"spinach", "color":"green", "classification":"leafy-green" }
{ "create" : { "_index" : "veggies", "_id" : "9" } }
{ "name":"arugula", "color":"green", "classification":"leafy-green" }
{ "create" : { "_index" : "veggies", "_id" : "10" } }
{ "name":"endive", "color":"green", "classification":"leafy-green" }
{ "create" : { "_index" : "veggies", "_id" : "11" } }
{ "name":"lettuce", "color":"green", "classification":"leafy-green" }
```

Ya puede buscar con un orden. Esta solicitud agrega un orden ascendente por clasificación:

```
GET veggies/_search
{
```

```
  "query" : {
    "term": { "color": "green" }
  },
  "sort" : [
      "classification"
  ]
}
```

Recursos relacionados

Para obtener más información, consulte los siguientes recursos:

- Introducción
- Indexación de datos
- Buscar datos

Tutorial: Migración a Amazon OpenSearch Service

Las instantáneas de índice son una forma popular de migrar desde un clúster de OpenSearch autoadministrado o Elasticsearch heredado a Amazon OpenSearch Service. En términos generales, el proceso consta de los siguientes pasos:

1. Tome una instantánea del clúster existente y cargue la instantánea en un bucket de Amazon S3.
2. Cree un dominio de OpenSearch Service.
3. Conceda permisos a OpenSearch Service para acceder al bucket, y asegúrese de disponer de permisos para trabajar con instantáneas.
4. Restaure la instantánea en el dominio de OpenSearch Service.

Este tutorial proporciona pasos más detallados y opciones alternativas, cuando corresponda.

Tomar y cargar la instantánea

Aunque puede utilizar el complemento repository-s3 para tomar instantáneas directamente a S3, tiene que instalar el complemento en cada nodo, ajustar `opensearch.yml` (o `elasticsearch.yml` si usa un clúster Elasticsearch), reiniciar cada nodo, agregar sus credenciales de AWS y, finalmente, tomar la instantánea. El complemento es una gran opción para uso continuo o para migrar clústeres más grandes.

En clústeres más pequeños, un enfoque único consiste en tomar una instantánea del sistema de archivos compartida y, luego, utilizar la AWS CLI para cargarla a S3. Si ya tiene una instantánea, vaya al paso 4.

Cómo tomar una instantánea y cargarla a Amazon S3

1. Agregue la configuración `path.repo` a `opensearch.yml` (o `Elasticsearch.yml`) en todos los nodos y, a continuación, reinicie cada nodo.

   ```
   path.repo: ["/my/shared/directory/snapshots"]
   ```

2. Registre un repositorio de instantáneas, necesario antes de tomar una instantánea. Un repositorio es solo una ubicación de almacenamiento: un sistema de archivos compartido, Amazon S3, un sistema de archivos distribuido de Hadoop (HDFS), etc. En este caso, utilizaremos un sistema de archivos compartido (“fs”):

   ```
   PUT _snapshot/my-snapshot-repo-name
   {
     "type": "fs",
     "settings": {
       "location": "/my/shared/directory/snapshots"
     }
   }
   ```

3. Tomar la instantánea:

   ```
   PUT _snapshot/my-snapshot-repo-name/my-snapshot-name
   {
     "indices": "migration-index1,migration-index2,other-indices-*",
     "include_global_state": false
   }
   ```

4. Instale la AWS CLI y ejecute `aws configure` para agregar sus credenciales.

5. Desplácese hasta el directorio de instantáneas. A continuación, ejecute los siguientes comandos para crear un nuevo bucket de S3 y cargar el contenido del directorio de instantáneas en ese bucket:

   ```
   aws s3 mb s3://amzn-s3-demo-bucket --region us-west-2
   aws s3 sync . s3://amzn-s3-demo-bucket --sse AES256
   ```

En función del tamaño de la instantánea y la velocidad de su conexión a internet, esta operación puede tardar un tiempo.

Crear un dominio

Aunque la consola es la forma más fácil de crear un dominio, en este caso, ya tiene el terminal abierto y la AWS CLI instalada. Modifique el siguiente comando para crear un dominio que se ajuste a sus necesidades:

```
aws opensearch create-domain \
  --domain-name migration-domain \
  --engine-version OpenSearch_1.0 \
  --cluster-config InstanceType=c5.large.search,InstanceCount=2 \
  --ebs-options EBSEnabled=true,VolumeType=gp2,VolumeSize=100 \
  --node-to-node-encryption-options Enabled=true \
  --encryption-at-rest-options Enabled=true \
  --domain-endpoint-options EnforceHTTPS=true,TLSSecurityPolicy=Policy-Min-
TLS-1-2-2019-07 \
  --advanced-security-options
 Enabled=true,InternalUserDatabaseEnabled=true,MasterUserOptions='{MasterUserName=master-
user,MasterUserPassword=master-user-password}' \
  --access-policies '{"Version": "2012-10-17",       "Statement":
[{"Effect":"Allow","Principal": {"AWS": "arn:aws:iam::aws-
region:user/UserName"},"Action":["es:ESHttp*"],"Resource":"arn:aws:es:aws-
region:111122223333:domain/migration-domain/*"}]}' \
  --region aws-region
```

Tal como está, el comando crea un dominio accesible a Internet con dos nodos de datos, cada uno con 100 GiB de almacenamiento. También habilita el control de acceso detallado con autenticación básica HTTP y toda la configuración de cifrado. Utilice la consola de OpenSearch Service si necesita una configuración de seguridad más avanzada, como una VPC.

Antes de ejecutar el comando, cambie el nombre de dominio, las credenciales del usuario maestro y el número de cuenta. Especifique la misma Región de AWS que utilizó para el bucket de S3 y una versión de OpenSearch/Elasticsearch que sea compatible con la instantánea.

> **Important**
>
> Las instantáneas solo son compatibles con versiones posteriores y solo con una versión principal. Por ejemplo, no puede restaurar una instantánea desde un clúster OpenSearch

1.x en un clúster Elasticsearch 7.x, solo un clúster OpenSearch 1.x o 2.x. La versión secundaria también importa. No se puede restaurar una instantánea desde un clúster 5.3.3 autoadministrado en un dominio de OpenSearch Service 5.3.2. Recomendamos elegir la versión más reciente de OpenSearch o Elasticsearch que admita la instantánea. Para obtener una tabla de versiones compatibles, consulte the section called “Uso de una instantánea para migrar datos”.

Conceda permisos al bucket de S3.

En la consola AWS Identity and Access Management (IAM), cree un rol con los siguientes permisos y relación de confianza. Al crear un rol, elija S3 como el Servicio de AWS. Asigne al rol el nombre `OpenSearchSnapshotRole` para que sea fácil de encontrar.

Permisos

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [{
      "Action": [
        "s3:ListBucket"
      ],
      "Effect": "Allow",
      "Resource": [
        "arn:aws:s3:::amzn-s3-demo-bucket"
      ]
    },
    {
      "Action": [
        "s3:GetObject",
        "s3:PutObject",
        "s3:DeleteObject"
      ],
      "Effect": "Allow",
      "Resource": [
        "arn:aws:s3:::amzn-s3-demo-bucket/*"
      ]
    }
  ]
```

```
}
```

Relación de confianza

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [{
      "Effect": "Allow",
      "Principal": {
        "Service": "es.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

Después, conceda permisos a su rol de IAM personal para asumir `OpenSearchSnapshotRole`. Cree la siguiente política y adjúntela a su identidad.

Permisos

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [{
      "Effect": "Allow",
      "Action": "iam:PassRole",
      "Resource": "arn:aws:iam::123456789012:role/OpenSearchSnapshotRole"
    }
  ]
}
```

Asigne el rol de instantánea en OpenSearch Dashboards (si utiliza el control de acceso detallado)

Si ha habilitado el control de acceso detallado, incluso si utiliza autenticación HTTP básica para todos los demás fines, debe asignar el rol `manage_snapshots` al rol de IAM para poder trabajar con instantáneas.

Para otorgar permisos a la identidad a fin de trabajar con instantáneas

1. Inicie sesión en el panel con las credenciales de usuario maestro que especificó al crear el dominio de OpenSearch Service. Puede encontrar la URL del panel en la consola de OpenSearch Service. Adopta el formato `https://`*`domain-endpoint`*`/_dashboards/`.
2. En el menú principal, seleccione Seguridad, Roles y seleccione el rol manage_snapshots.
3. Seleccione Usuarios asignados, Administrar mapeo.
4. Agregue el ARN del dominio de su rol de IAM personal en el campo apropiado. El ARN está en uno de los siguientes formatos:

   ```
   arn:aws:iam::123456789123:user/user-name
   ```

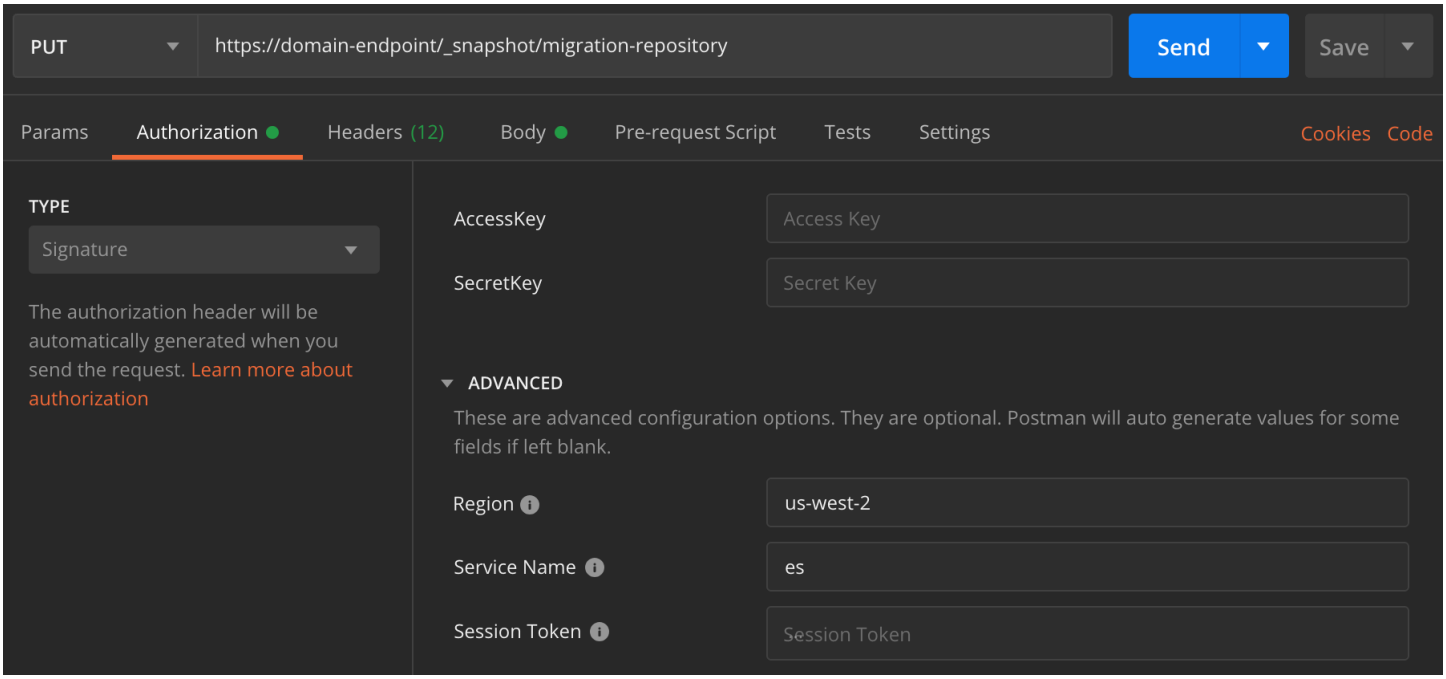
   ```
   arn:aws:iam::123456789123:role/role-name
   ```

5. Seleccione Asignar y confirme que el rol aparece en Usuarios asignados.

Restaurar la instantánea

En este punto, tiene dos formas de acceder a su dominio de OpenSearch Service: la autenticación básica HTTP con sus credenciales de usuario maestro o la autenticación de AWS con las credenciales de IAM. Dado que las instantáneas utilizan Amazon S3, que no tiene un concepto de usuario maestro, debe utilizar sus credenciales de IAM para registrar el repositorio de instantáneas en su dominio de OpenSearch Service.

La mayoría de los lenguajes de programación tienen bibliotecas para ayudar con las solicitudes de firma, pero el enfoque más simple es utilizar una herramienta como Postman y poner sus credenciales de IAM en la sección Autorización.



Para restaurar la instantánea

1. Independientemente de cómo elija firmar sus solicitudes, el primer paso es registrar el repositorio:

   ```
   PUT _snapshot/my-snapshot-repo-name
   {
     "type": "s3",
     "settings": {
       "bucket": "amzn-s3-demo-bucket",
       "region": "us-west-2",
       "role_arn": "arn:aws:iam::123456789012:role/OpenSearchSnapshotRole"
     }
   }
   ```

2. A continuación, enumere las instantáneas en el repositorio y busque la que desea restaurar. En este punto, puede continuar utilizando Postman o cambiar a una herramienta como curl.

   Abreviado

   ```
   GET _snapshot/my-snapshot-repo-name/_all
   ```

   curl

```
curl -XGET -u 'master-user:master-user-password' https://domain-endpoint/
_snapshot/my-snapshot-repo-name/_all
```

3. Restaurare la instantánea.

   Abreviado

   ```
   POST _snapshot/my-snapshot-repo-name/my-snapshot-name/_restore
   {
     "indices": "migration-index1,migration-index2,other-indices-*",
     "include_global_state": false
   }
   ```

   curl

   ```
   curl -XPOST -u 'master-user:master-user-password' https://domain-endpoint/
   _snapshot/my-snapshot-repo-name/my-snapshot-name/_restore \
     -H 'Content-Type: application/json' \
     -d '{"indices":"migration-index1,migration-index2,other-indices-
   *","include_global_state":false}'
   ```

4. Finalmente, verifique que sus índices se restablecieron como se esperaba.

   Abreviado

   ```
   GET _cat/indices?v
   ```

   curl

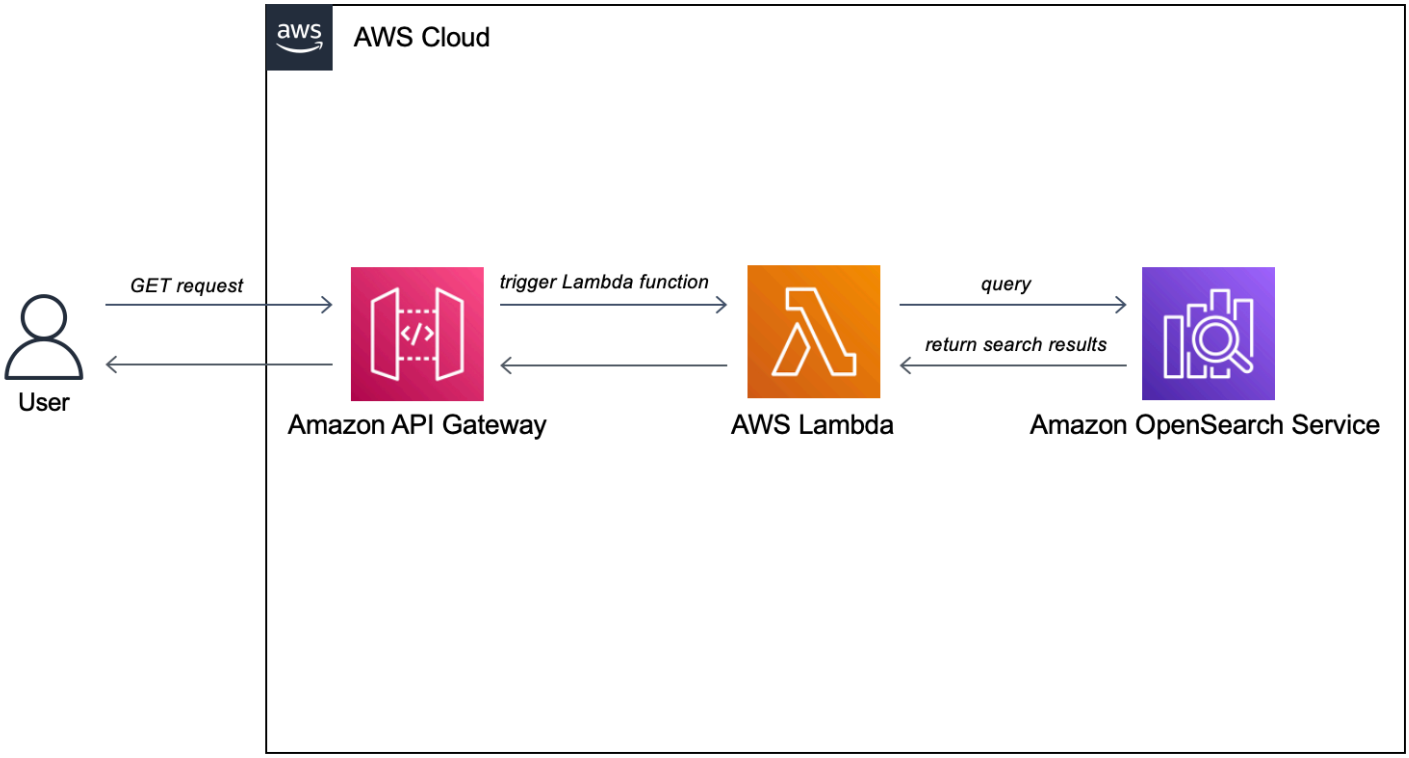
   ```
   curl -XGET -u 'master-user:master-user-password' https://domain-endpoint/_cat/
   indices?v
   ```

En este punto, la migración está completa. Puede configurar sus clientes para que utilicen el nuevo punto de conexión de OpenSearch Service, cambiar el tamaño del dominio para adaptarse a su carga de trabajo, comprobar el recuento de particiones de sus índices, cambiar a un usuario maestro de IAM o comenzar a crear visualizaciones en OpenSearch Dashboards.

Tutorial: Creación de una aplicación de búsqueda con Amazon OpenSearch Service

Una forma común para crear una aplicación de búsqueda con Amazon OpenSearch Service es utilizar formularios web para enviar consultas de usuarios a un servidor. A continuación, puede autorizar el servidor para llamar a las API de OpenSearch Service directamente y que el servidor envíe solicitudes a OpenSearch Service. Pero si desea escribir un código del lado del cliente que no dependa de un servidor, debe compensar los riesgos de seguridad y rendimiento. Permitir acceso público sin firma a las API de OpenSearch es inadmisible. Los usuarios pueden acceder a puntos de conexión no protegidos o influir en el rendimiento del clúster a través de consultas demasiado complejas (o demasiadas consultas).

Este capítulo se presenta una solución: utilizar Amazon API Gateway para restringir a los usuarios un subconjunto de API de OpenSearch y que AWS Lambda firme solicitudes de API Gateway a OpenSearch Service.



> **Note**
>
> Se aplican precios estándar de API Gateway y Lambda, pero dentro del uso limitado de este tutorial, los costos deben ser insignificantes.

Requisitos previos

Un requisito previo para este tutorial es un dominio de OpenSearch Service. Si todavía no tiene uno, siga los pasos de Creación de un dominio de OpenSearch Service para crear uno.

Paso 1: Indexe los datos de muestra

Descargue sample-movies.zip, descomprímalo y utilice la operación _bulk de la API para agregar 5000 documentos al índice `movies`:

```
POST https://search-my-domain.us-west-1.es.amazonaws.com/_bulk
{ "index": { "_index": "movies", "_id": "tt1979320" } }
{"directors":["Ron
 Howard"],"release_date":"2013-09-02T00:00:00Z","rating":8.3,"genres":
["Action","Biography","Drama","Sport"],"image_url":"http://ia.media-imdb.com/images/
M/MV5BMTQyMDE0MTY0OV5BMl5BanBnXkFtZTcwMjI2OTI0OQ@@._V1_SX400_.jpg","plot":"A re-
creation of the merciless 1970s rivalry between Formula One rivals James Hunt and
 Niki Lauda.","title":"Rush","rank":2,"running_time_secs":7380,"actors":["Daniel
 Brühl","Chris Hemsworth","Olivia Wilde"],"year":2013,"id":"tt1979320","type":"add"}
{ "index": { "_index": "movies", "_id": "tt1951264" } }
{"directors":["Francis Lawrence"],"release_date":"2013-11-11T00:00:00Z","genres":
["Action","Adventure","Sci-Fi","Thriller"],"image_url":"http://ia.media-imdb.com/
images/M/
MV5BMTAyMjQ3OTAxMzNeQTJeQWpwZ15BbWU4MDU0NzA1MzAx._V1_SX400_.jpg","plot":"Katniss
 Everdeen and Peeta Mellark become targets of the Capitol after
 their victory in the 74th Hunger Games sparks a rebellion in
 the Districts of Panem.","title":"The Hunger Games: Catching
 Fire","rank":4,"running_time_secs":8760,"actors":["Jennifer Lawrence","Josh
 Hutcherson","Liam Hemsworth"],"year":2013,"id":"tt1951264","type":"add"}
...
```

Tenga en cuenta que lo anterior es un comando de ejemplo con un pequeño subconjunto de datos disponibles. Para realizar la operación `_bulk`, debe copiar y pegar todo el contenido del archivo `sample-movies`. Para obtener más instrucciones, consulte the section called “Opción 2: cargar varios documentos”.

También puede utilizar el siguiente comando curl para conseguir el mismo resultado:

```
curl -XPOST -u 'master-user:master-user-password' 'domain-endpoint/_bulk' --data-binary
 @bulk_movies.json -H 'Content-Type: application/json'
```

Paso 2: crear e implementar una función de Lambda

Antes de crear su API en API Gateway, cree la función de Lambda a la que pasa las solicitudes.

Crear la función de Lambda

En esta solución, API Gateway pasa solicitudes a una función de Lambda, que consulta OpenSearch Service y devuelve los resultados. Dado que esta función de ejemplo utiliza bibliotecas externas, debe crear un paquete de implementación y cargarlo en Lambda.

Creación del paquete de implementación

1. Abra un símbolo del sistema y cree un directorio del proyecto `my-opensearch-function`. Por ejemplo, en macOS:

   ```
   mkdir my-opensearch-function
   ```

2. Desplácese hasta el directorio del proyecto `my-sourcecode-function`.

   ```
   cd my-opensearch-function
   ```

3. Copie el contenido del siguiente código Python de ejemplo y guárdelo en un nuevo archivo denominado `opensearch-lambda.py`. Añada su región y el punto de conexión del host al archivo.

   ```
   import boto3
   import json
   import requests
   from requests_aws4auth import AWS4Auth

   region = '' # For example, us-west-1
   service = 'es'
   credentials = boto3.Session().get_credentials()
   awsauth = AWS4Auth(credentials.access_key, credentials.secret_key, region, service,
    session_token=credentials.token)
   ```

```
host = '' # The OpenSearch domain endpoint with https:// and without a trailing
 slash
index = 'movies'
url = host + '/' + index + '/_search'

# Lambda execution starts here
def lambda_handler(event, context):

    # Put the user query into the query DSL for more accurate search results.
    # Note that certain fields are boosted (^).
    query = {
        "size": 25,
        "query": {
            "multi_match": {
                "query": event['queryStringParameters']['q'],
                "fields": ["title^4", "plot^2", "actors", "directors"]
            }
        }
    }

    # Elasticsearch 6.x requires an explicit Content-Type header
    headers = { "Content-Type": "application/json" }

    # Make the signed HTTP request
    r = requests.get(url, auth=awsauth, headers=headers, data=json.dumps(query))

    # Create the response and add some extra content to support CORS
    response = {
        "statusCode": 200,
        "headers": {
            "Access-Control-Allow-Origin": '*'
        },
        "isBase64Encoded": False
    }

    # Add the search results to the response
    response['body'] = r.text
    return response
```

4. Instale las bibliotecas externas en un nuevo directorio `package`.

```
pip3 install --target ./package boto3
pip3 install --target ./package requests
```

```
pip3 install --target ./package requests_aws4auth
```

5. Cree un paquete de implementación con las bibliotecas instaladas en la raíz. El siguiente comando genera un archivo `my-deployment-package.zip` en el directorio de su proyecto.

```
cd package
zip -r ../my-deployment-package.zip .
```

6. Agregue el archivo `opensearch-lambda.py` a la raíz del archivo zip.

```
cd ..
zip my-deployment-package.zip opensearch-lambda.py
```

Para obtener más información sobre cómo crear funciones de Lambda y paquetes de implementación, consulte Deploy Python Lambda functions with .zip file archives en la Guía para desarrolladores de AWS Lambda y the section called “Crear el paquete de implementación de Lambda” en esta guía.

Cómo crear la función usando la consola de Lambda

1. Navegue hasta la consola de Lambda en https://console.aws.amazon.com/lambda/home. En el panel de navegación izquierdo, seleccione Funciones.
2. Seleccione Crear función.
3. Configure los siguientes campos:

   - Nombre de función: opensearch-function
   - Tiempo de ejecución: Python 3.9
   - Arquitectura: x86_64

   Mantenga todas las demás opciones predeterminadas y elija Crear función.
4. En la sección Código fuente de la página de resumen de la función, seleccione el menú desplegable Cargar desde y seleccione el archivo .zip. Busque el archivo `my-deployment-package.zip` que ha creado y seleccione Guardar.
5. El controlador es el método de su código de función que procesa eventos. En Configuración de tiempo de ejecución, seleccione Editar y cambie el nombre del controlador de acuerdo con el nombre del archivo en el paquete de implementación donde se encuentra la función de Lambda. Como el archivo se denomina `opensearch-lambda.py`, cambie el nombre del controlador

a *opensearch-lambda*.lambda_handler. Para más información, consulte Controlador de funciones de Lambda en Python.

Paso 3: crear la API en API Gateway

Usar API Gateway le permite crear una API más limitada y simplifica el proceso de interactuar con la API _search de OpenSearch. API Gateway le permite habilitar las características de seguridad, como la autenticación de Amazon Cognito y la limitación controlada. Realice los siguientes pasos para crear e implementar una API:

Crear y configurar la API

Para crear una API mediante la consola de API Gateway

1. Navegue hasta la consola de API Gateway en https://console.aws.amazon.com/apigateway/home. En el panel de navegación izquierdo, seleccione APIs.
2. Localice API de REST (no privado) y elija Crear.
3. En la siguiente página, busque la sección Crear nueva API y asegúrese de que Nueva API esté seleccionada.
4. Configure los siguientes campos:
   - Nombre de la API: opensearch-api
   - Descripción: API pública para buscar un dominio de Amazon OpenSearch Service
   - Tipo de punto de conexión: Regional
5. Seleccione Crear API.
6. Seleccione Acciones y Crear método.
7. Seleccione GET (Obtener) en el menú desplegable y haga clic en la marca de verificación para confirmar.
8. Configure los siguientes ajustes y, a continuación, elija Guardar:

Opción	Valor
Tipo de integración	Función de Lambda
Usar integración de proxy de Lambda	Sí

Opción	Valor
Región de Lambda	*us-west-1*
Función de Lambda	opensearch-lambda
Usar tiempo de espera predeterminado	Sí

Configuración de la solicitud de método

Seleccione Solicitud de método y configure los siguientes ajustes:

Opción	Valor
Autorización	NONE
Validador de solicitud	Validar parámetros de cadena de consulta y encabezados
Clave de API requerida	falso

En Parámetros de cadena de consulta de URL, seleccione Añadir cadena de consulta y configure el siguiente parámetro:

Opción	Valor
Nombre	q
Obligatorio	Sí

Implemente la API y configure una etapa

La consola de API Gateway le permite implementar una API creando una implementación y asociándola a una etapa nueva o una existente.

1. Seleccione Acciones e Implementar API.
2. Para Etapa de implementación elija Nueva etapa y nombre la etapa opensearch-api-test.

3. Seleccione Implementar.
4. Configure los siguientes ajustes en el editor de etapas y, a continuación, elija Guardar los cambios:

Opción	Valor
Habilitar limitación controlada	Sí
Rate (Tarifa)	1 000
Ráfagas	500

Estos ajustes configuran una API que solo tiene un método: una solicitud `GET` al punto de conexión raíz (`https://`*`some-id`*`.execute-api.`*`us-west-1`*`.amazonaws.com/search-es-api-test`). La solicitud requiere un solo parámetro (q), la cadena de consulta que se busca. Cuando se llama, el método pasa la solicitud a Lambda, que ejecuta la función `opensearch-lambda`. Para más información, consulte Creación de una API en Amazon API Gateway e Implementación de una API de REST en Amazon API Gateway.

Paso 4: (opcional) modificar la política de acceso a dominios

Su dominio de OpenSearch Service debe permitir a la función de Lambda realizar solicitudes `GET` en el índice `movies`. Si su dominio tiene una política de acceso abierto con control de acceso detallado habilitado, puede dejarlo tal cual:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
      },
      "Action": "es:*",
      "Resource": "arn:aws:es:us-west-1:123456789012:domain/domain-name/*"
    }
```

```
  ]
}
```

Como alternativa, puede elegir hacer su política de acceso a dominio más detallada. Por ejemplo, la siguiente política mínima proporciona acceso de lectura a `opensearch-lambda-role` (creado a través de Lambda) al índice `movies`. Para obtener el nombre exacto del rol que Lambda crea automáticamente, vaya a la consola de AWS Identity and Access Management (IAM), elija Roles, y busque “Lambda”.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::123456789012:role/service-role/opensearch-lambda-role-1abcdefg"
      },
      "Action": "es:ESHttpGet",
      "Resource": "arn:aws:es:us-west-1:123456789012:domain/domain-name/movies/_search"
    }
  ]
}
```

> **Important**
>
> Si tiene habilitado un control de acceso detallado para el dominio, también deberá que asignar el rol a un usuario en OpenSearch Dashboards; de lo contrario, verá errores de permisos.

Configure los permisos del rol de ejecución de Lambda

Además de configurar la política de acceso al dominio, también debe asegurarse de que la función de ejecución de Lambda tenga los permisos de IAM necesarios para acceder a su dominio de

OpenSearch Service. La función de Lambda requiere permisos específicos en función de si se utiliza un dominio administrado o una colección OpenSearch Service sin servidor.

Para los dominios administrados de OpenSearch Service:

Adjunte la siguiente política de IAM a su función de ejecución de Lambda para que pueda realizar solicitudes a su dominio de OpenSearch Service:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "es:ESHttpGet",
        "es:ESHttpPost"
      ],
      "Resource": "arn:aws:es:us-west-1:123456789012:domain/domain-name/*"
    }
  ]
}
```

Para las colecciones sin servidor de OpenSearch Service:

Si utiliza OpenSearch Service sin servidor, adjunte la siguiente política de IAM a su función de ejecución de Lambda:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "aoss:*",
      "Resource": "arn:aws:aoss:us-west-1:123456789012:collection/collection-id"
    }
  ]
```

```
}
```

Para adjuntar estas políticas a su rol de ejecución de Lambda:

1. Desplácese a la consola de IAM en https://console.aws.amazon.com/iam/.
2. Elija Funciones y busque su función de ejecución de Lambda (normalmente denominada `opensearch-lambda-role-xxxxxxxx`).
3. Elija Agregar permisos y luego Crear política insertada.
4. Elija la pestaña JSON y pegue la política correspondiente de arriba, sustituyendo los valores de los marcadores de posición por los ARN de sus recursos reales.
5. Elija Revisar política, escriba un nombre como `OpenSearchAccess` y seleccione Crear política.

> **Note**
>
> Sin estos permisos de IAM, su función de Lambda recibirá errores de "Acceso denegado" al intentar consultar su dominio de OpenSearch Service, incluso si la política de acceso al dominio permite las solicitudes.

Para obtener más información sobre las políticas de acceso, consulte the section called "Configuración de políticas de acceso".

Asigne el rol de Lambda (si utiliza un control de acceso detallado)

El control de acceso detallado presenta un paso adicional antes de poder probar la aplicación. Incluso si utiliza la autenticación básica HTTP para todos los demás fines, es necesario asignar el rol de Lambda a un usuario, de lo contrario verá errores de permisos.

1. Desplácese hasta la URL de OpenSearch Dashboards para el dominio.
2. En el menú principal, seleccione Seguridad, Roles y elija el enlace a `all_access`, el rol al que necesita asignar el rol de Lambda.
3. Seleccione Usuarios asignados, Administrar mapeo.
4. En Roles de backend, agregue el nombre de recurso de Amazon (ARN) del rol de Lambda. El ARN debe adoptar la forma de `arn:aws:iam::123456789123:role/service-role/opensearch-lambda-role-1abcdefg`.

5. Seleccione Asignar y confirme que el usuario o el rol aparecen en Usuarios asignados.

Paso 5: Pruebe la aplicación web

Para probar la aplicación web

1. Descargue sample-site.zip, descomprímalo y abra `scripts/search.js` en el editor de texto de su elección.
2. Actualice la variable `apigatewayendpoint` para que apunte al punto de conexión de API Gateway y añada una barra oblicua al final de la ruta en cuestión. Puede encontrar rápidamente el punto de conexión en API Gateway seleccionando Etapas y seleccionando el nombre de la API. La variable `apigatewayendpoint` debe adoptar la forma de `https://`*`some-id`*`.execute-api.`*`us-west-1`*`.amazonaws.com/opensearch-api-test/`.
3. Abra `index.html` e intente ejecutar búsquedas de thor, house y algunos otros términos.

Movie Search

thor

Found 7 results.



Thor

2011 — The powerful but arrogant god Thor is cast out of Asgard to live amongst humans in Midgard (Earth), where he soon becomes one of their finest defenders.



Thor: The Dark World

2013 — Faced with an enemy that even Odin and Asgard cannot withstand, Thor must embark on his most perilous and personal journey yet, one that will reunite him with Jane Foster and force him to sacrifice everything to save us all.



Vikingdom

2013 — A forgotten king, Eirick, is tasked with the impossible odds to defeat Thor, the God of Thunder.

Solucionar errores de CORS

Aunque la función de Lambda incluye contenido en la respuesta para admitir CORS, es posible que aparezca el siguiente error:

```
Access to XMLHttpRequest at '<api-gateway-endpoint>' from origin 'null' has been
 blocked by CORS policy: No 'Access-Control-Allow-Origin' header is present in the
 requested resource.
```

Si esto sucede, pruebe lo siguiente:

1. Habilitar CORS en el recurso GET. En Avanzado, configure Access-Control-Allow-Credentials en `'true'`.
2. Vuelva a implementar su API en API Gateway (Acciones, Implementar API).
3. Elimine y vuelva a agregar el activador de la función de Lambda. Agregue volver a añadirlo, seleccione Agregar desencadenador y cree el punto de conexión HTTP que invoca su función. El desencadenador debe tener la siguiente configuración:

Desencadenador	API	Etapa de implementación	Seguridad
API Gateway	opensearch-api	opensearch-api-test	Open

Pasos a seguir a continuación

Este capítulo es solo un punto de partida para demostrar un concepto. Podría considerar las siguientes modificaciones:

- Agregue sus propios datos al dominio de OpenSearch Service.
- Agregue métodos a su API.
- En la función de Lambda, modifique la consulta de búsqueda o aumente los diferentes campos.
- Aplique un estilo diferente a los resultados o modifique `search.js` para mostrar diferentes campos al usuario.

Tutorial: Visualización de llamadas de soporte al cliente con OpenSearch Service y OpenSearch Dashboards

Este capítulo es un tutorial completo de la siguiente situación: el servicio de atención al cliente de una empresa recibe llamadas y quiere analizarlas. ¿Cuál es el tema de cada llamada? ¿Cuántas fueron positivas? ¿Cuántas negativas? ¿Cómo los gerentes pueden buscar o revisar las transcripciones de estas llamadas?

Un flujo de trabajo manual puede implicar que los empleados escuchen las grabaciones, apunten el tema de cada llamada y decidan si la interacción con el cliente fue positiva.

Dicho proceso requeriría mucho esfuerzo y trabajo. Suponiendo un tiempo medio de 10 minutos por llamada, cada empleado podría escuchar solo 48 llamadas al día. Salvo sesgo humano, los datos que generan sería de alta precisión, pero la cantidad de datos sería mínima: tan solo el tema de la llamada y un valor booleano para determinar si el cliente quedó satisfecho. Cualquier información más compleja, como, por ejemplo, una transcripción completa, requeriría gran cantidad de tiempo.

El uso de Amazon S3, Amazon Transcribe, Amazon Comprehend y Amazon OpenSearch Service, puede automatizar un proceso similar con muy poco código y obtener muchos más datos. Por ejemplo, puede obtener una transcripción completa de la llamada, palabras clave de la transcripción, y un "sentimiento" general de la llamada (positiva, negativa, neutra o mixta). A continuación, puede utilizar OpenSearch y OpenSearch Dashboards para buscar y visualizar los datos.

Aunque puede utilizar este tutorial tal y como está, su finalidad es dar ideas sobre cómo enriquecer sus documentos JSON antes de indexarlos en OpenSearch Service.

Costo estimado

En general, seguir los pasos que se indican en este tutorial debería costar menos de 2 USD. En este tutorial se utilizan los siguientes recursos:

- bucket de S3 con menos de 100 MB transferidos y almacenados

  Para obtener más información, consulte Precios de Amazon S3.
- Un dominio de OpenSearch Service con una instancia `t2.medium` y 10 GiB de almacenamiento de EBS durante varias horas

  Para obtener más información, consulte Precios de Amazon OpenSearch Service.
- Varias llamadas a Amazon Transcribe

Para obtener más información, consulte Precios de Amazon Transcribe.

- Varias llamadas de procesamiento de lenguaje natural a Amazon Comprehend

  Para obtener más información, consulte Precios de Amazon Comprehend.

Temas

- Paso 1: configure los requisitos previos
- Paso 2: copie el código de muestra
- (Opcional) Paso 3: indexe los datos de ejemplo
- Paso 4: analice y visualice sus datos
- Paso 5: elimine recursos y pasos siguientes

Paso 1: configure los requisitos previos

Antes de continuar, debe contar con los siguientes recursos.

Requisito previo	Descripción
Bucket de Amazon S3	Para obtener más información, consulte Creación de un bucket en la Guía del usuario de Amazon Simple Storage Service.
Dominio de OpenSearch Service	El destino de los datos. Para obtener más información, consulte Creación de dominios de OpenSearch Service.

Si aún no tiene estos recursos, puede crearlos utilizando los siguientes comandos de la AWS CLI:

```
aws s3 mb s3://my-transcribe-test --region us-west-2
```

```
aws opensearch create-domain --domain-name my-transcribe-test --engine-version
 OpenSearch_1.0 --cluster-config  InstanceType=t2.medium.search,InstanceCount=1
 --ebs-options EBSEnabled=true,VolumeType=standard,VolumeSize=10 --access-policies
 '{"Version": "2012-10-17",       "Statement":[{"Effect":"Allow","Principal":
{"AWS":"arn:aws:iam::123456789012:root"},"Action":"es:*","Resource":"arn:aws:es:us-
west-2:123456789012:domain/my-transcribe-test/*"}]}' --region us-west-2
```

> **Note**
>
> Estos comandos utilizan la región `us-west-2`, pero puede utilizar cualquier región que admita Amazon Comprehend. Para obtener más información, consulte Referencia general de AWS.

Paso 2: copie el código de muestra

1. Copie y pegue el siguiente código de muestra de Python 3 en un archivo nuevo denominado `call-center.py`:

```
import boto3
import datetime
import json
import requests
from requests_aws4auth import AWS4Auth
import time
import urllib.request

# Variables to update
audio_file_name = '' # For example, 000001.mp3
bucket_name = '' # For example, my-transcribe-test
domain = '' # For example, https://search-my-transcribe-test-12345.us-
west-2.es.amazonaws.com
index = 'support-calls'
type = '_doc'
region = 'us-west-2'

# Upload audio file to S3.
s3_client = boto3.client('s3')

audio_file = open(audio_file_name, 'rb')

print('Uploading ' + audio_file_name + '...')
response = s3_client.put_object(
    Body=audio_file,
    Bucket=bucket_name,
    Key=audio_file_name
)

# # Build the URL to the audio file on S3.
```

```
# # Only for the us-east-1 region.
# mp3_uri = 'https://' + bucket_name + '.s3.amazonaws.com/' + audio_file_name

# Get the necessary details and build the URL to the audio file on S3.
# For all other regions.
response = s3_client.get_bucket_location(
    Bucket=bucket_name
)
bucket_region = response['LocationConstraint']
mp3_uri = 'https://' + bucket_name + '.s3-' + bucket_region + '.amazonaws.com/' +
 audio_file_name

# Start transcription job.
transcribe_client = boto3.client('transcribe')

print('Starting transcription job...')
response = transcribe_client.start_transcription_job(
    TranscriptionJobName=audio_file_name,
    LanguageCode='en-US',
    MediaFormat='mp3',
    Media={
        'MediaFileUri': mp3_uri
    },
    Settings={
        'ShowSpeakerLabels': True,
        'MaxSpeakerLabels': 2 # assumes two people on a phone call
    }
)

# Wait for the transcription job to finish.
print('Waiting for job to complete...')
while True:
    response =
 transcribe_client.get_transcription_job(TranscriptionJobName=audio_file_name)
    if response['TranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED',
 'FAILED']:
        break
    else:
        print('Still waiting...')
    time.sleep(10)

transcript_uri = response['TranscriptionJob']['Transcript']['TranscriptFileUri']

# Open the JSON file, read it, and get the transcript.
```

```
response = urllib.request.urlopen(transcript_uri)
raw_json = response.read()
loaded_json = json.loads(raw_json)
transcript = loaded_json['results']['transcripts'][0]['transcript']

# Send transcript to Comprehend for key phrases and sentiment.
comprehend_client = boto3.client('comprehend')

# If necessary, trim the transcript.
# If the transcript is more than 5 KB, the Comprehend calls fail.
if len(transcript) > 5000:
    trimmed_transcript = transcript[:5000]
else:
    trimmed_transcript = transcript

print('Detecting key phrases...')
response = comprehend_client.detect_key_phrases(
    Text=trimmed_transcript,
    LanguageCode='en'
)

keywords = []
for keyword in response['KeyPhrases']:
    keywords.append(keyword['Text'])

print('Detecting sentiment...')
response = comprehend_client.detect_sentiment(
    Text=trimmed_transcript,
    LanguageCode='en'
)

sentiment = response['Sentiment']

# Build the Amazon OpenSearch Service URL.
id = audio_file_name.strip('.mp3')
url = domain + '/' + index + '/' + type + '/' + id

# Create the JSON document.
json_document = {'transcript': transcript, 'keywords': keywords, 'sentiment':
 sentiment, 'timestamp': datetime.datetime.now().isoformat()}

# Provide all details necessary to sign the indexing request.
credentials = boto3.Session().get_credentials()
```

```
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key, region,
 'opensearchservice', session_token=credentials.token)

# Index the document.
print('Indexing document...')
response = requests.put(url, auth=awsauth, json=json_document, headers=headers)

print(response)
print(response.json())
```

2. Actualice las seis variables iniciales.
3. Instale los paquetes necesarios mediante los siguientes comandos:

```
pip install boto3
pip install requests
pip install requests_aws4auth
```

4. Guarde el MP3 en el mismo directorio que `call-center.py` y ejecute el script. A continuación se indica un ejemplo de salida:

```
$ python call-center.py
Uploading 000001.mp3...
Starting transcription job...
Waiting for job to complete...
Still waiting...
Still waiting...
Still waiting...
Still waiting...
Still waiting...
Still waiting...
Still waiting...
Detecting key phrases...
Detecting sentiment...
Indexing document...
<Response [201]>
{u'_type': u'call', u'_seq_no': 0, u'_shards': {u'successful': 1, u'failed': 0,
 u'total': 2}, u'_index': u'support-calls4', u'_version': 1, u'_primary_term': 1,
 u'result': u'created', u'_id': u'000001'}
```

`call-center.py` realiza una serie de operaciones:

1. El script carga un archivo de audio (en este caso, un MP3, pero Amazon Transcribe es compatible con varios formatos) en su bucket de S3.
2. Envía la URL del archivo de audio a Amazon Transcribe y espera que realice la tarea de transcripción.

   El tiempo necesario para terminar el trabajo de transcripción depende de la duración del archivo de audio. Supongamos que tomará minutos, no segundos.

   > **Tip**
   >
   > Para mejorar la calidad de la transcripción, puede configurar un vocabulario personalizado para Amazon Transcribe.

3. Una vez que el trabajo de transcripción termina, el script extrae la transcripción, la recorta a 5 000 caracteres y envía a Amazon Comprehend para identificar palabras clave y analizar el sentimiento.
4. Por último, el script añade toda la transcripción, palabras clave, sentimiento y marca de tiempo actual a un documento JSON, y lo indexa en OpenSearch Service.

> **Tip**
>
> LibriVox tiene audiolibros de dominio público que puede utilizar para hacer pruebas.

(Opcional) Paso 3: indexe los datos de ejemplo

Si no tiene a mano varias grabaciones de llamadas, y no es algo muy frecuente, puede indexar los documentos de muestra en sample-calls.zip, que son comparables a lo que produce `call-center.py`.

1. Cree un archivo denominado `bulk-helper.py`:

```
import boto3
from opensearchpy import OpenSearch, RequestsHttpConnection
import json
from requests_aws4auth import AWS4Auth

host = '' # For example, my-test-domain.us-west-2.es.amazonaws.com
```

```
region = '' # For example, us-west-2
service = 'es'

bulk_file = open('sample-calls.bulk', 'r').read()

credentials = boto3.Session().get_credentials()
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key, region, service,
 session_token=credentials.token)

search = OpenSearch(
    hosts = [{'host': host, 'port': 443}],
    http_auth = awsauth,
    use_ssl = True,
    verify_certs = True,
    connection_class = RequestsHttpConnection
)

response = search.bulk(bulk_file)
print(json.dumps(response, indent=2, sort_keys=True))
```

2. Actualice las dos variables iniciales para `host` y `region`.
3. Instale el paquete necesario mediante el siguiente comando:

```
pip install opensearch-py
```

4. Descargue y descomprima sample-calls.zip.
5. Guarde `sample-calls.bulk` en el mismo directorio que `bulk-helper.py` y ejecute el ayudante. A continuación se indica un ejemplo de salida:

```
$ python bulk-helper.py
{
  "errors": false,
  "items": [
    {
      "index": {
        "_id": "1",
        "_index": "support-calls",
        "_primary_term": 1,
        "_seq_no": 42,
        "_shards": {
          "failed": 0,
          "successful": 1,
```

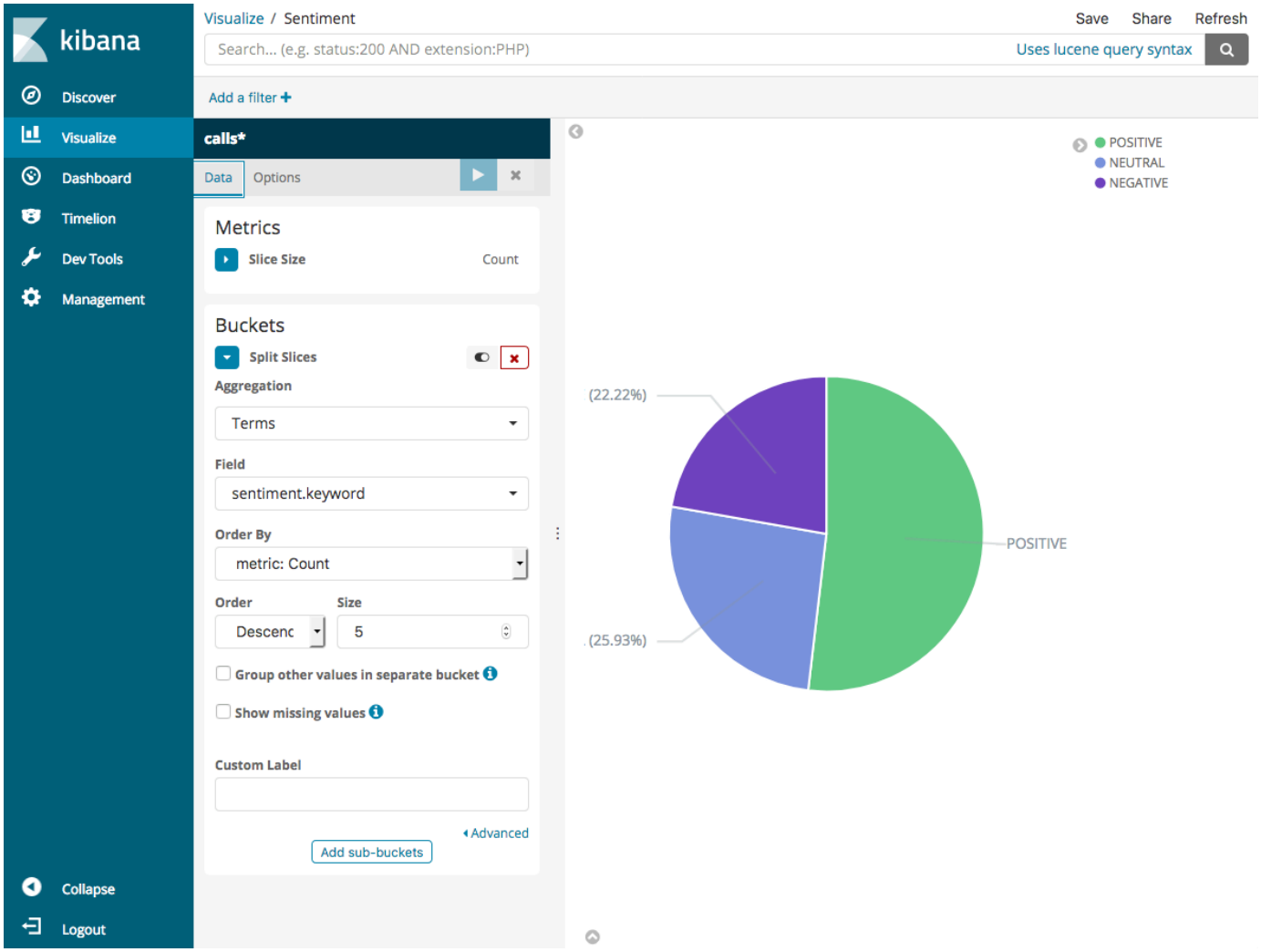
```
          "total": 2
        },
        "_type": "_doc",
        "_version": 9,
        "result": "updated",
        "status": 200
      }
    },
    ...
  ],
  "took": 27
}
```

Paso 4: analice y visualice sus datos

Ahora que tiene en algunos datos en OpenSearch Service, puede visualizarlos con OpenSearch Dashboards.

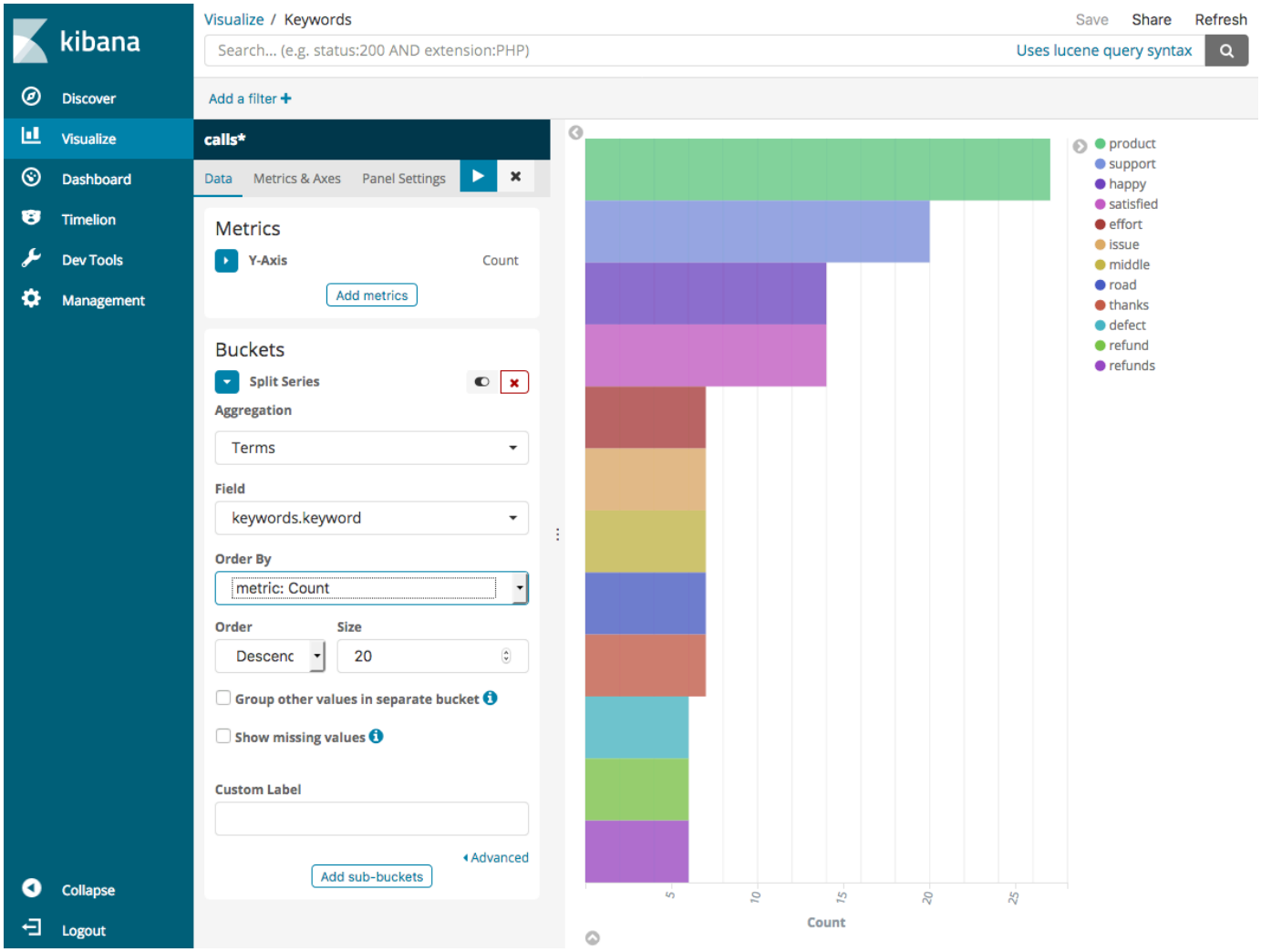
1. Vaya a `https://search-`*`domain`*`.`*`region`*`.es.amazonaws.com/_dashboards`.
2. Para poder utilizar OpenSearch Dashboards, necesita un patrón de índice. Dashboards utiliza patrones de índice para limitar su análisis a uno o más índices. Para que coincida con el `support-calls`índice que`call-center.py` cree, diríjase a Pila de administración, Patrones, y defina un patrón de índice de `support*`. y luego elija Paso siguiente.
3. En Time Filter field name (Nombre de campo Filtro de tiempo), elija timestamp (marca temporal).
4. Ahora pude comenzar a crear visualizaciones. Seleccione Visualize (Visualizar) y añada una nueva visualización.
5. Seleccione el gráfico circular y el patrón de índice `support*`.
6. El valor predeterminado de visualización es básico. Si quiere crear una visualización más interesante, elija Split Slices (Sectores divididos).

   Para Aggregation (Agregación), elija Terms (Términos). En Field (Campo), elija sentiment.keyword. Seleccione Apply changes (Aplicar cambios) y Save (Guardar).

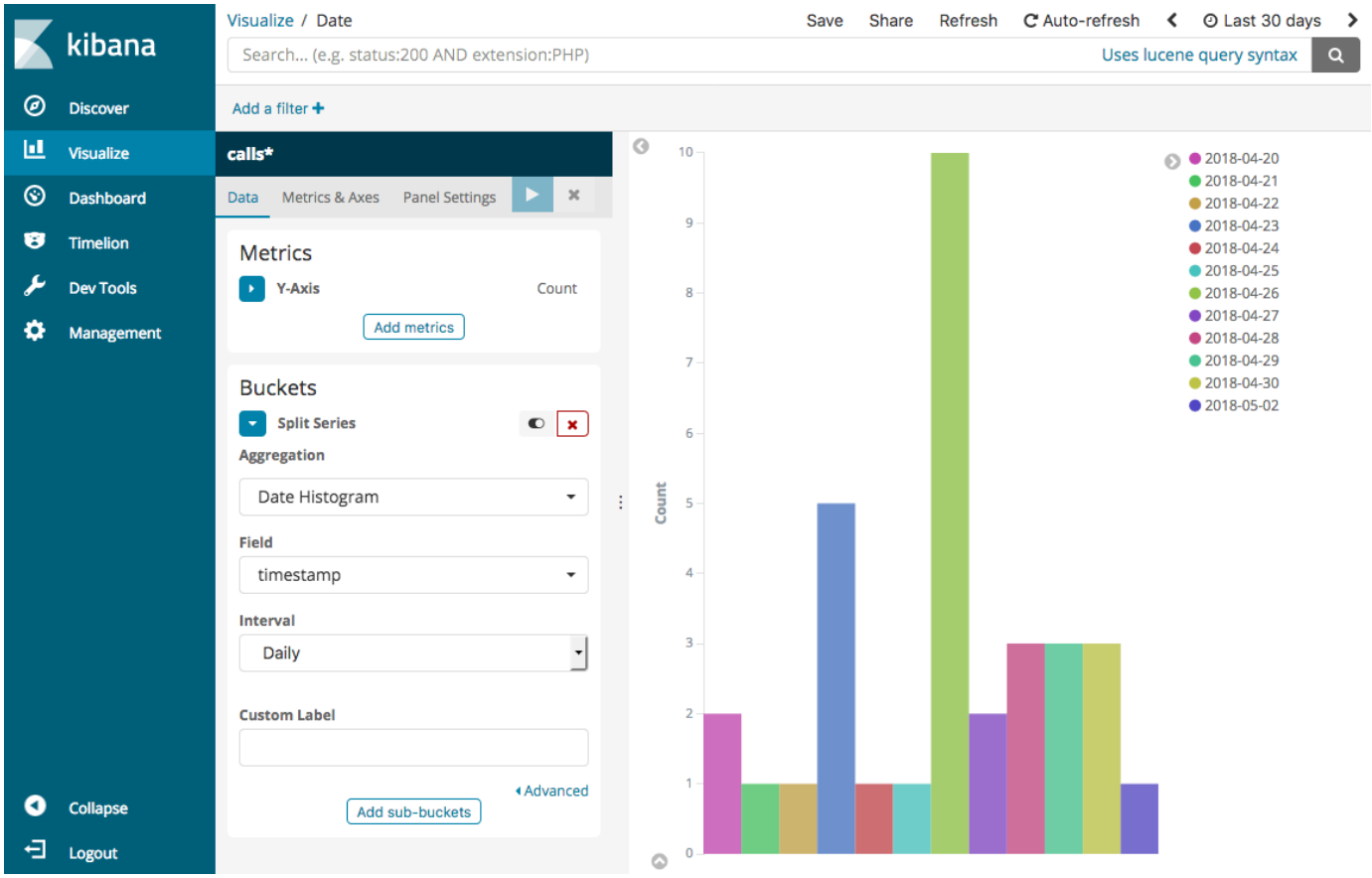


7. Vuelva a la página Visualizar y agregue otra visualización. Esta vez, elija el gráfico de barras horizontales.
8. Seleccione Split Series (Series divididas).

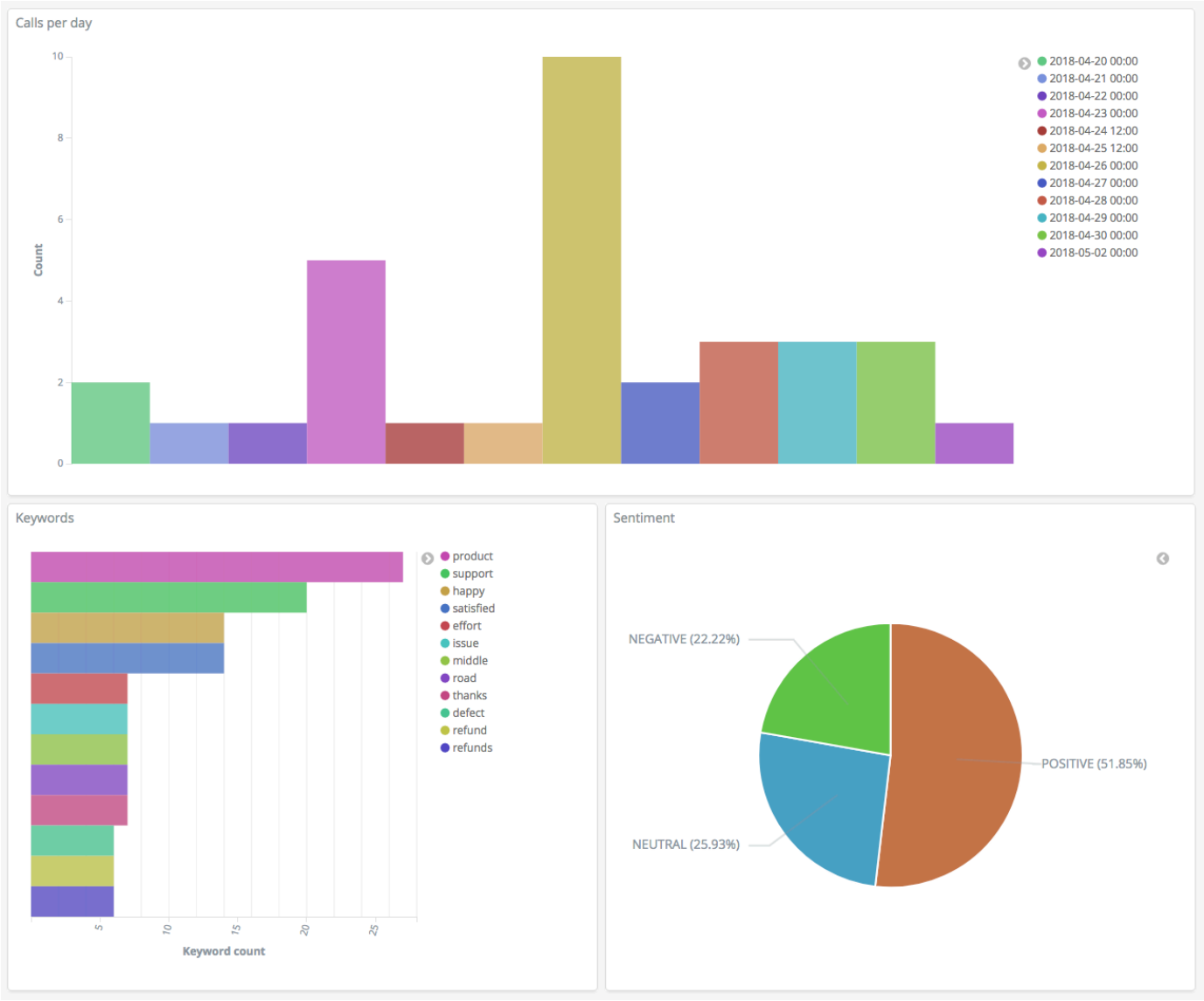
   Para Aggregation (Agregación), elija Terms (Términos). En Field (Campo), elija keywords.keyword y cambie el valor de Size (Tamaño) a 20. Seleccione Apply Changes (Aplicar cambios) y Save (Guardar).



9. Vuelva a la página Visualizar y agregue una última visualización, un gráfico de barras verticales.
10. Seleccione Split Series (Series divididas). En Aggregation (Agregación), elija Date Histogram (Histograma de fecha). En Field (Campo), elija timestamp (marca temporal) y cambie Interval (Intervalo) a Daily (Diario).
11. Seleccione Metrics & Axes (Métricas y ejes) y cambie Mode (Modo) a normal.
12. Seleccione Apply Changes (Aplicar cambios) y Save (Guardar).



13. Ahora que tiene tres visualizaciones, puede añadirlas a un panel de Dashboards. Seleccione Dashboard (Panel), cree un panel y añada sus visualizaciones.



Paso 5: elimine recursos y pasos siguientes

Para evitar gastos innecesarios, elimine el bucket de S3 y el dominio de OpenSearch Service. Para obtener más información, consulte Eliminación de un bucket en la Guía del usuario de Amazon Simple Storage Service y Eliminación de un dominio de OpenSearch Service en esta guía.

Las transcripciones requieren mucho menos espacio de disco que los archivos MP3. Puede reducir su ventana de retención de MP3 por ejemplo, de tres meses de grabaciones de llamadas a un mes guarde años de transcripciones, y ahorre en costos de almacenamiento.

También puede automatizar el proceso de transcripción con AWS Step Functions y Lambda, agregue los metadatos adicionales antes de indexar, o cree visualizaciones más complejas para adaptarse a su caso de uso exacto.

Cambio de nombre de Amazon OpenSearch Service: resumen de cambios

El 8 de septiembre de 2021, nuestro conjunto de aplicaciones de búsqueda y análisis pasó a llamarse Amazon OpenSearch Service. OpenSearch Service es compatible con OpenSearch, así como con Elasticsearch OSS heredado. En las secciones siguientes, se describen las diferentes partes del servicio que sufrieron modificaciones con el cambio de nombre y las acciones que se deben realizar para garantizar que los dominios sigan funcionando correctamente.

Algunos de estos cambios solo se aplican cuando actualiza sus dominios de Elasticsearch a OpenSearch. En otros casos, como en la consola de Billing and Cost Management, la experiencia cambia inmediatamente.

Tenga en cuenta que esta lista no es exhaustiva. Mientras que otras partes del producto también cambiaron, estas actualizaciones son las más relevantes.

Nueva versión de la API

La nueva versión de la API de configuración de OpenSearch Service (01/01/2021) funciona con OpenSearch, así como con Elasticsearch OSS heredado. Se reemplazaron 21 operaciones de la API por nombres más concisos e independientes del motor (por ejemplo, `CreateElasticsearchDomain` cambió a `CreateDomain`), pero OpenSearch Service aún es compatible con ambas versiones de la API.

Recomendamos utilizar las nuevas operaciones de la API para crear y administrar dominios en el futuro. Tenga en cuenta que cuando utilice las nuevas operaciones de la API para crear un dominio, debe especificar el parámetro `EngineVersion` en el formato `Elasticsearch_X.Y` o `OpenSearch_X.Y`, en lugar de solo el número de versión. Si no especifica ninguna versión, toma de forma predeterminada la versión más reciente de OpenSearch.

Actualice la AWS CLI a la versión 1.20.40 o posterior para utilizar `aws opensearch ...` a fin de crear y administrar los dominios. Para obtener el nuevo formato de la CLI, consulte la Referencia de la CLI de OpenSearch.

Tipos de instancia con cambio de nombre

Los tipos de instancias de Amazon OpenSearch Service ahora tienen el formato `<type>.<size>.search`, por ejemplo, `m6g.large.search` en lugar de `m6g.large.elasticsearch`. No es necesario realizar ninguna acción. Los dominios existentes comenzarán a hacer referencia automáticamente a los nuevos tipos de instancias dentro de la API y en la consola de Billing and Cost Management.

Si dispone de instancias reservadas (RI), el cambio no afectará al contrato. La versión antigua de la API de configuración aún es compatible con el formato de nomenclatura anterior, pero si desea utilizar la nueva versión de la API, debe utilizar el nuevo formato.

Cambios en las políticas de acceso

En las secciones siguientes, se describen las acciones que debe realizar para actualizar las políticas de acceso.

Políticas de IAM

Recomendamos que actualice sus Políticas de IAM para utilizar las operaciones de la API con cambio de nombre. Sin embargo, OpenSearch Service seguirá respetando las políticas existentes al replicar internamente los permisos de la API antiguos. Por ejemplo, si actualmente tiene permiso para realizar la operación `CreateElasticsearchDomain`, ahora puede realizar llamadas a ambos `CreateElasticsearchDomain` (operación antigua de la API) y `CreateDomain` (nueva operación de la API). Lo mismo se aplica a las denegaciones explícitas. Para obtener una lista de las operaciones de la API actualizadas, consulte la referencia de elementos de política.

Políticas de SCP

Las políticas de control de servicios (SCP) presentan una capa adicional de complejidad en comparación con IAM estándar. Para evitar que las políticas de SCP se rompan, debe agregar tanto las operaciones de la API antiguas como las nuevas a cada una de sus políticas de SCP. Por ejemplo, si actualmente un usuario tiene permisos para `CreateElasticsearchDomain`, también debe concederles permisos para `CreateDomain` a fin de que puedan retener la capacidad de crear dominios. Lo mismo se aplica a las denegaciones explícitas.

Por ejemplo:

```
"Statement": [
```

```
    {
      "Effect": "Allow",
      "Action": [
        "es:CreateElasticsearchDomain",
        "es:CreateDomain"
         ...
      ],
    },
      "Effect": "Deny",
      "Action:" [
        "es:DeleteElasticsearchDomain",
        "es:DeleteDomain"
         ...
```

Nuevos tipos de recursos

OpenSearch Service presenta los siguientes tipos de recursos nuevos:

Recurso	Descripción
`AWS::OpenSearchService::Domain`	Representa un dominio de Amazon OpenSearch Service. Este recurso existe en el nivel de servicio y no es específico del software que se ejecuta en el dominio. Se aplica a servicios como CloudFormation y AWS Resource Groups, en los que crea y administra recursos para el servicio en su conjunto. Para obtener instrucciones para actualizar los dominios definidos en CloudFormation de Elasticsearch a OpenSearch, consulte Observaciones en la Guía del usuario de CloudFormation.
`AWS::OpenSearch::Domain`	Representa el software OpenSearch/Elasticsearch que se ejecuta en un dominio. Este recurso se aplica a servicios como AWS CloudTrail y AWS Config, que hacen referenci a al software que se ejecuta en el dominio

Recurso	Descripción
	en lugar de a OpenSearch Service como un todo. Estos servicios ahora contienen tipos de recursos separados para dominios que ejecutan Elasticsearch (`AWS::Elasticsearch::Domain`) frente a dominios que ejecutan OpenSearch (`AWS::OpenSearch::Domain`).

> **Note**
>
> En AWS Config, seguirá viendo sus datos en el tipo de recurso `AWS::Elasticsearch::Domain` existente durante varias semanas, incluso si actualiza uno o más dominios a OpenSearch.

Kibana cambió de nombre a OpenSearch Dashboards

OpenSearch Dashboards, la alternativa de AWS a Kibana, es una herramienta de visualización de código abierto diseñada para funcionar con OpenSearch. Después de actualizar un dominio de Elasticsearch a OpenSearch, el punto de conexión `/_plugin/kibana` cambia a `/_dashboards`. OpenSearch Service redirigirá todas las solicitudes al nuevo punto de enlace, pero si utiliza el punto de enlace de Kibana en cualquiera de sus políticas de IAM, actualice esas políticas para incluir el nuevo punto de enlace de `/_dashboards` también.

Si utiliza the section called “Autenticación SAML para OpenSearch Dashboards”, antes de actualizar su dominio a OpenSearch, debe cambiar todas las URL de Kibana configuradas en su proveedor de identidad (IdP) de `/_plugin/kibana` a `/_dashboards`. Las URL más comunes son las URL de servicio al consumidor de aserción (ACS) y las URL de destinatario.

Se ha cambiado el nombre del rol `kibana_read_only` predeterminado para OpenSearch Dashboards por `opensearch_dashboards_read_only`, y al rol `kibana_user` se le ha cambiado el nombre por `opensearch_dashboards_user`. El cambio es de aplicación a todos los dominios recién creados de OpenSearch 1.x en los que se ejecute el software de servicio R20211203 o posterior. Si se actualiza un dominio existente al software de servicio R20211203, los nombres de los roles siguen siendo los mismos.

Métricas de CloudWatch con cambio de nombre

Varias métricas de CloudWatch cambian para los dominios que ejecutan OpenSearch. Cuando actualiza un dominio a OpenSearch, las métricas cambian automáticamente y las alarmas actuales de CloudWatch fallarán. Antes de actualizar el clúster de una versión de Elasticsearch a una versión de OpenSearch, asegúrese de actualizar las alarmas de CloudWatch para utilizar las nuevas métricas.

Cambiaron las siguientes métricas:

Nombre de métrica original	Nombre nuevo
KibanaHealthyNodes	OpenSearchDashboardsHealthyNodes
KibanaConcurrentConnections	OpenSearchDashboardsConcurrentConnections
KibanaHeapTotal	OpenSearchDashboardsHeapTotal
KibanaHeapUsed	OpenSearchDashboardsHeapUsed
KibanaHeapUtilization	OpenSearchDashboardsHeapUtilization
KibanaOS1MinuteLoad	OpenSearchDashboardsOS1MinuteLoad
KibanaRequestTotal	OpenSearchDashboardsRequestTotal
KibanaResponseTimesMaxInMillis	OpenSearchDashboardsResponseTimesMaxInMillis
ESReportingFailedRequestSysErrCount	KibanaReportingFailedRequestSysErrCount
ESReportingRequestCount	KibanaReportingRequestCount
ESReportingFailedRequestUserErrCount	KibanaReportingFailedRequestUserErrCount

Nombre de métrica original	Nombre nuevo
`ESReportingSuccessCount`	`KibanaReportingSuccessCount`
`ElasticsearchRequests`	`OpenSearchRequests`

Para obtener una lista completa de las métricas que OpenSearch Service envía a Amazon CloudWatch, consulte the section called “Monitoreo de métricas del clúster”.

Cambios en la consola de Billing and Cost Management

Los datos históricos en la consola de Administración de facturación y costos y en los Informes de costos y usos seguirán utilizando el nombre del servicio antiguo, por lo que debe comenzar a emplear filtros tanto para Amazon OpenSearch Service como para el nombre de Elasticsearch heredado cuando haga búsquedas de datos. Si ya tiene informes guardados, actualice los filtros para asegurarse de que también incluyen OpenSearch Service. Es posible que reciba inicialmente una alerta cuando su utilización disminuya para Elasticsearch y aumente para OpenSearch, pero desaparezca en varios días.

Además del nombre del servicio, los siguientes campos cambiarán para todos los informes, facturas y operaciones de la API de lista de precios:

Campo	Formato antiguo	Formato nuevo
Tipo de instancia	`m5.large.elasticsearch`	`m5.large.search`
Familia de productos	Instancia de Elasticsearch Volumen de Elasticsearch	Instancia de Amazon OpenSearch Service Volumen de Amazon OpenSearch Service
Descripción del precio	5,098 USD por hora de instancia c5.18xlarge.elasticsearch (u hora parcial), UE	5,098 USD por hora de instancia c5.18xlarge.search (u hora parcial), UE

Campo	Formato antiguo	Formato nuevo
Familia de instancias	`ultrawarm.elasticsearch`	`ultrawarm.search`

Nuevo formato de evento

El formato de los eventos que OpenSearch Service envía a Amazon EventBridge y Amazon CloudWatch ha cambiado, específicamente el campo `detail-type`. El campo fuente (`aws.es`) aún es el mismo. Para obtener el formato completo de cada tipo de evento, consulte the section called "Supervisión de eventos". Si tiene reglas de evento existentes que dependen del formato antiguo, asegúrese de actualizarlas para que se ajusten al nuevo formato.

¿Qué permanece igual?

Las siguientes características y funcionalidades, entre otras no enumeradas, permanecerán iguales:

- Entidad principal de servicio (`es.amazonaws.com`)
- Código de proveedor
- ARN del dominio
- Puntos de conexión del dominio

Comience: Actualice sus dominios a OpenSearch 1.x

OpenSearch 1.x soporta actualizaciones de Elasticsearch versión 6.8 y 7.x. Para obtener instrucciones a fin de actualizar su dominio, consulte the section called "Actualización de un dominio (consola)". Si utiliza la AWS CLI o la API de configuración para actualizar su dominio, debe especificar la `TargetVersion` como `OpenSearch_1.x`.

OpenSearch 1.x presenta una configuración de dominio adicional llamada Habilitar modo de compatibilidad. Debido a que algunos clientes y complementos de Elasticsearch OSS verifican la versión del clúster antes de conectarse, el modo de compatibilidad establece que OpenSearch informe su versión como 7.10 para que estos clientes sigan funcionando.

Puede habilitar el modo de compatibilidad al crear dominios de OpenSearch por primera vez o al actualizar a OpenSearch desde una versión de Elasticsearch. Si no está establecido, el parámetro predeterminado es `false` al crear un dominio y `true` cuando actualiza un dominio.

Para habilitar el modo de compatibilidad mediante la API de configuración, establezca `override_main_response_version` en `true`:

```
POST https://es.us-east-1.amazonaws.com/2021-01-01/opensearch/upgradeDomain
{
  "DomainName": "domain-name",
  "TargetVersion": "OpenSearch_1.0",
  "AdvancedOptions": {
    "override_main_response_version": "true"
   }
}
```

Para habilitar o desactivar el modo de compatibilidad en dominios de OpenSearch existentes, es necesario utilizar la operación de la API de OpenSearch _cluster/settings:

```
PUT /_cluster/settings
{
  "persistent" : {
    "compatibility.override_main_response_version" : true
  }
}
```

Solución de problemas de Amazon OpenSearch Service

En este tema se describe cómo identificar y resolver problemas comunes de Amazon OpenSearch Service. Consulte la información de esta sección antes de ponerse en contacto con el Soporte de AWS.

No se puede acceder a OpenSearch Dashboards

El punto de conexión de OpenSearch Dashboards no admite solicitudes firmadas. Si la política de control de acceso del dominio solo concede acceso a determinados roles de IAM y no se ha configurado la autenticación de Amazon Cognito, es posible que aparezca el siguiente error cuando se intente acceder a Dashboards:

```
"User: anonymous is not authorized to perform: es:ESHttpGet"
```

Si su dominio de OpenSearch Service utiliza el acceso mediante VPC, probablemente no ocurra este error, pero puede que se agote el tiempo de espera. Para obtener más información acerca de cómo solucionar este problema y las distintas opciones de configuración disponibles the section called “Control del acceso a Dashboards ”, consulte the section called “Acerca de las políticas de acceso en los dominios de VPC” y the section called “Identity and Access Management”.

No se puede obtener acceso al dominio de VPC

Consulte the section called “Acerca de las políticas de acceso en los dominios de VPC” y the section called “Probar los dominios de VPC”.

Clúster en estado de solo lectura

Comparado con versiones anteriores de Elasticsearch, OpenSearch y Elasticsearch 7.x utilizan un sistema diferente para la coordinación de clústeres. En este nuevo sistema, cuando el clúster pierde quórum, el clúster no estará disponible hasta que realice alguna acción. La pérdida de quórum puede adoptar dos formas:

- Si el clúster utiliza nodos principales dedicados, la pérdida de cuórum se produce cuando la mitad o más no están disponibles.

- Si el clúster no utiliza nodos principales dedicados, la pérdida de cuórum se produce cuando la mitad o más de los nodos de datos no están disponibles.

Si se produce una pérdida de cuórum y el clúster tiene más de un nodo, OpenSearch Service restaura el cuórum y coloca el clúster en un estado de solo lectura. Dispone de dos opciones para hacerlo:

- Elimine el estado de solo lectura y utilice el clúster tal y como está.
- Restaure el clúster o los índices individuales a partir de una instantánea.

Si prefiere utilizar el clúster tal y como está, verifique que el estado del clúster sea verde mediante la siguiente solicitud:

```
GET _cat/health?v
```

Si el estado del clúster es rojo, le recomendamos que restaure el clúster a partir de una instantánea. También puede consultar the section called “Estado rojo del clúster” para ver los pasos de solución de problemas. Si el estado del clúster es verde, compruebe que todos los índices esperados estén presentes utilizando la siguiente solicitud:

```
GET _cat/indices?v
```

A continuación, ejecute algunas búsquedas para verificar que los datos esperados están presentes. Si es, puede eliminar el estado de solo lectura utilizando la siguiente solicitud:

```
PUT _cluster/settings
{
  "persistent": {
    "cluster.blocks.read_only": false
  }
}
```

Si se produce una pérdida de cuórum y el clúster solo tiene un nodo, OpenSearch Service sustituye el nodo y no coloca el clúster en un estado de solo lectura. De lo contrario, las opciones son las mismas: utilice el clúster tal y como está o restaure a partir de una instantánea.

En ambas situaciones, OpenSearch Service envía dos eventos a su AWS Health Dashboard. El primero le informa de la pérdida de cuórum. El segundo ocurre después de que OpenSearch Service

restaure correctamente el cuórum. Para obtener más información acerca del uso de AWS Health Dashboard, consulte la Guía de usuario de AWS Health.

Estado rojo del clúster

El estado rojo del clúster significa que al menos una partición principal y sus réplicas no están asignados a un nodo. OpenSearch Service sigue tratando de tomar instantáneas automatizadas de todos los índices independientemente de su estado, pero las instantáneas fallan mientras el estado del clúster rojo persiste.

Las causas más comunes de un estado rojo del clúster son que los nodos del clúster han fallado y que el proceso de OpenSearch se ha bloqueado debido a una carga de procesamiento elevada continua.

> **Note**
>
> OpenSearch Service almacena instantáneas automatizadas durante 14 días, independientemente del estado del clúster. Por lo tanto, si el estado del clúster rojo persiste durante más de dos semanas, se eliminará la última instantánea automatizada correcta y podría perder permanentemente los datos del clúster. Si el dominio de OpenSearch Service pasa a un estado rojo del clúster, el equipo de Soporte podría contactar con usted para preguntarle si desea corregir el problema por sí mismo o si prefiere que ellos lo ayuden. También puede definir una alarma de CloudWatch que le notifique cuando se produzca un estado rojo del clúster.

En última instancia, las particiones rojas causan clústeres rojos y los índices rojos causan particiones rojas. Para identificar los índices que causan el estado rojo del clúster, OpenSearch tiene algunas API útiles.

- `GET /_cluster/allocation/explain` elige la primera partición no asignada que encuentra y explica por qué no se puede asignar a un nodo:

```
{
    "index": "test4",
    "shard": 0,
    "primary": true,
    "current_state": "unassigned",
    "can_allocate": "no",
```

```
    "allocate_explanation": "cannot allocate because allocation is not permitted to
 any of the nodes"
}
```

- `GET /_cat/indices?v` muestra el estado, el número de documentos y el uso de disco de cada índice:

```
health status index            uuid                   pri rep docs.count docs.deleted
 store.size pri.store.size
green  open   test1            30h1EiMvS5uAFr2t5CEVoQ   5   0        820            0
       14mb           14mb
green  open   test2            sdIxs_WDT56afFGu5KPbFQ   1   0          0            0
       233b           233b
green  open   test3            GGRZp_TBRZuSaZpAGk2pmw   1   1          2            0
     14.7kb          7.3kb
red    open   test4            BJxfAErbTtu5HBjIXJV_7A   1   0
green  open   test5            _8C6MIXOSxCqVYicH3jsEA   1   0          7            0
     24.3kb         24.3kb
```

Eliminar los índices rojos es la forma más rápida de solucionar el estado rojo del clúster. Según el motivo del estado rojo del clúster, podría ampliar el dominio de OpenSearch Service para utilizar tipos de instancias de mayor tamaño, más instancias o más almacenamiento basado en EBS e intentar volver a crear los índices problemáticos.

Si eliminar un índice problemático no es factible, puede restaurar una instantánea, eliminar documentos del índice, cambiar la configuración del índice, reducir el número de réplicas o eliminar otros índices para liberar espacio en el disco. El paso más importante es resolver el estado rojo del clúster antes de volver a configurar su dominio de OpenSearch Service. Volver a configurar un dominio con un estado rojo del clúster puede agravar el problema y el dominio podría quedarse bloqueado en el estado de configuración Processing (Procesando) hasta que se resuelva el estado.

Corrección automática de clústeres en rojo

Si el estado del clúster permanece en rojo de manera continuada durante más de una hora, OpenSearch Service intenta corregirlo automáticamente redirigiendo particiones no asignadas o restaurando desde instantáneas anteriores.

Si no consigue corregir uno o varios índices en rojo y el estado del clúster permanece en rojo durante un total de 14 días, OpenSearch Service solo toma medidas adicionales si el clúster cumple al menos uno de los siguientes criterios:

- Tiene solo una zona de disponibilidad
- No tiene nodos maestros dedicados
- Contiene tipos de instancias de ráfaga (T2 o T3)

En este momento, si el clúster cumple uno de estos criterios, OpenSearch Service envía notificaciones diarias durante los siguientes 7 días en las que explica que, si no se corrigen estos índices, se eliminarán todas las particiones no asignadas. Si el estado del clúster sigue en rojo transcurridos 21 días, OpenSearch Service elimina las particiones no asignadas (almacenamiento y computación) de todos los índices en rojo. Las notificaciones correspondientes a cada uno de estos eventos se reciben en el panel Notificaciones de la consola de OpenSearch Service. Para obtener más información, consulte the section called “Eventos del estado del clúster”.

Recuperación de una carga de procesamiento elevada continua

Para determinar si el estado rojo de un clúster se debe a una carga de procesamiento elevada continua en un nodo de datos, monitorice las siguientes métricas de clúster.

Métrica relevante	Descripción	Recuperación
JVMMemoryPressure	Especifica el porcentaje del montón de Java que se emplea para todos los nodos de datos de un clúster. Vea la estadística Máximo de esta métrica y busque caídas cada vez más pequeñas en la presión de memoria a medida que el recolector de elementos no utilizados de Java no consigue reclamar suficiente memoria. Probablemente, este patrón se deba a consultas complejas o a campos de datos de gran tamaño. Los tipos de instancias x86 utilizan el recolector de basura Concurrent Mark Sweep (CMS), que se ejecuta junto a subprocesos de aplicaciones	Establezca interruptores de memoria para la JVM. Para obtener más información, consulte the section called “OutOfMemoryError de JVM”. Si el problema persiste, elimine los índices innecesarios, reduzca el número o la complejidad de las solicitudes al dominio, agregue instancias o utilice tipos de instancia más grandes.

Métrica relevante	Descripción	Recuperación
	para que las pausas sigan siendo breves. Si CMS no puede obtener memoria suficiente durante sus recolecciones normales, desencadena una recolección de basura completa, lo que puede provocar pausas prolongadas de las aplicaciones y afectar a la estabilidad del clúster. Los tipos de instancias Graviton basados en ARM utilizan el recolector de basura Garbage-First (G1), que es similar a CMS, pero utiliza pausas breves adicionales y desfragmentación del montón para reducir aún más la necesidad de recolecciones de basura completas. En cualquier caso, si el uso de memoria continúa creciendo por encima de lo que el recopilador de basura puede obtener durante la recopilación de basura completa, OpenSearch se bloquea con un error de memoria insuficiente. En todos los tipos de instancias, una buena regla es mantener el uso por debajo del 80 %. La API `_nodes/stats/jvm` ofrece un resumen útil sobre estadísticas de JVM, el uso del grupo de memoria e información de recopilación de elementos no utilizados:	

Métrica relevante	Descripción	Recuperación
	`GET domain-endpoint /_nodes/stats/jvm?pretty`	
CPUUtilization	Especifica el porcentaje de recursos de CPU usados para los nodos de datos de un clúster. Consulte la estadística Máximo para esta métrica y busque un patrón continuo de uso elevado.	Agregue nodos de datos o aumente el tamaño de los tipos de instancia de los nodos de datos existentes.
Nodos	Especifica el número de nodos de un clúster. Vea la estadística Mínimo para esta métrica. Este valor fluctúa cuando el servicio implementa una nueva flota de instancias para un clúster.	Agregue nodos de datos.

Estado amarillo del clúster

Un estado amarillo del clúster significa que los fragmentos principales de todos los índices están asignados a nodos de un clúster, pero los fragmentos de réplica de al menos un índice no lo están. Los clústeres de un nodo siempre se inicializan con un estado amarillo, ya que no existe ningún otro nodo al que OpenSearch Service pueda asignar una réplica. Para conseguir el estado verde del clúster, aumente el número de nodos. Para obtener más información, consulte the section called “Determinación del tamaño de dominios”.

Los clústeres de varios nodos pueden tener brevemente un estado de clúster amarillo después de crear un nuevo índice o después de un error de nodo. Este estado se resuelve automáticamente a medida que OpenSearch replica datos en todo el clúster. La falta de espacio en disco también puede causar el estado de clúster amarillo; el clúster sólo puede distribuir particiones de réplica si los nodos tienen espacio en disco para acomodarlos.

ClusterBlockException

Puede que reciba un error `ClusterBlockException` por los siguientes motivos.

Falta de espacio de almacenamiento disponible

Si uno o más nodos de su clúster tienen un espacio de almacenamiento inferior al valor mínimo de 1) 20 % del espacio de almacenamiento disponible, o 2) 20 GiB de espacio de almacenamiento, las operaciones básicas de escritura, como la adición de documentos y la creación de índices, pueden empezar a fallar. the section called “Cálculo de requisitos de almacenamiento” proporciona un resumen de cómo OpenSearch Service utiliza el espacio en disco.

Para evitar problemas, monitorice la métrica `FreeStorageSpace` en la consola de OpenSearch Service y cree alarmas de CloudWatch que se activan cuando `FreeStorageSpace` esté por debajo de un determinado umbral. `GET /_cat/allocation?v` también proporciona un resumen útil de la asignación de particiones y el uso del disco. Para solucionar los problemas asociados a una falta de espacio de almacenamiento, amplíe su dominio de OpenSearch Service para utilizar tipos de instancias de mayor tamaño, más instancias o más almacenamiento basado en EBS.

Presión alta de memoria de JVM

Si la métrica JVMMemoryPressure supera el 92 % durante 30 minutos, OpenSearch Service activa un mecanismo de protección y bloquea todas las operaciones de escritura para impedir que el clúster alcance el estado rojo. Cuando la protección está activada, se produce un error `ClusterBlockException` en las operaciones de escritura, los nuevos índices no se pueden crear y se genera el error `IndexCreateBlockException`.

Si la métrica JVMMemoryPressure vuelve al 88 % o un valor inferior durante cinco minutos, la protección queda deshabilitada y las operaciones de escritura en el clúster se desbloquean.

La presión alta de memoria de JVM puede deberse a picos en el número de solicitudes al clúster, asignaciones de particiones desequilibradas en los nodos, demasiadas particiones en un clúster, explosiones de mapeo de índices o datos de campo, o tipos de instancias que no pueden gestionar las cargas entrantes. También puede deberse al uso de agregaciones, comodines o intervalos de tiempo amplios en las consultas.

Para reducir el tráfico al clúster y resolver problemas de presión alta de memoria de JVM, pruebe una o más de las siguientes acciones:

- Escale el dominio para que el tamaño máximo de la pila por nodo sea de 32 GB.
- Reduzca el número de particiones mediante la eliminación de los índices antiguos o no utilizados.

- Borre la memoria caché de datos con la Operación de la API de `POST index-name/_cache/clear?fielddata=true`. Tenga en cuenta que borrar la memoria caché puede interrumpir las consultas en curso.

En general, para evitar una presión alta de memoria de JVM en el futuro, siga estas prácticas recomendadas:

- Evite la agregación en los campos de texto o cambie el tipo de mapeo para sus índices `keyword`.
- Optimice las solicitudes de búsqueda e indexación mediante la selección del número correcto de particiones.
- Configure políticas de Index State Management (ISM) paraeliminar los índices que no se utilizan periódicamente.

Error al migrar a multi-AZ con modo de espera

Se pueden producir los siguientes problemas al migrar un dominio existente a Multi-AZ con modo de espera.

Creación de un índice, una plantilla de índice o una política de ISM durante la migración de dominios sin modo de espera a dominios con modo de espera

Si crea un índice al migrar un dominio de Multi-AZ sin modo de espera a uno con modo de espera, y la plantilla de índice o la política de ISM no siguen las pautas de copia de datos recomendadas, esto puede provocar una incoherencia en los datos y es posible que la migración no se realice correctamente. Para evitar esta situación, cree el nuevo índice con un recuento de copias de datos (incluidos los nodos principales y las réplicas) que sea múltiplo de tres. Puede comprobar el progreso de la migración mediante la API de `DescribeDomainChangeProgress`. Si encuentra un error en el recuento de réplicas, corrija el error y, a continuación, póngase en contacto con Soporte de AWS para volver a intentar la migración.

Número incorrecto de copias de datos

Si no tiene el número correcto de copias de datos en su dominio, la migración a Multi-AZ con modo de espera fallará.

OutOfMemoryError de JVM

`OutOfMemoryError` de una JVM significa normalmente que se ha alcanzado uno de los siguientes interruptores de la JVM.

Interruptor	Descripción	Propiedad de configuración del clúster
Interruptor principal	Porcentaje total de memoria del montón de la JVM permitido para todos los interruptores. El valor predeterminado es 95 %.	`indices.breaker.total.limit`
Interruptor de datos de campo	Porcentaje de memoria del montón de la JVM permitido para cargar en memoria un único campo de datos. El valor predeterminado es 40 %. Si carga datos con campos de gran tamaño, probablemente deba aumentar este límite.	`indices.breaker.fielddata.limit`
Interruptor de solicitud	Porcentaje de memoria del montón de la JVM permitido para las estructuras de datos que se usan para responder a una solicitud de servicio. El valor predeterminado es 60 %. Si sus solicitud es de servicio implican el cálculo de agregaciones, es posible que deba aumentar este límite.	`indices.breaker.request.limit`

Nodos de clúster defectuosos

Las instancias de Amazon EC2 pueden experimentar terminaciones y reinicios inesperados. Normalmente, OpenSearch Service reinicia los nodos automáticamente. Sin embargo, es posible que uno o varios nodos de un clúster de OpenSearch permanezcan en una condición de error.

Para comprobar si se da esta condición, abra el panel del dominio en la consola de OpenSearch Service. Vaya a la pestaña Estado del clúster y busque la métrica Nodos totales. Compruebe si el número de nodos que aparece es menor que el número que configuró para el clúster. Si la métrica indica que uno o varios nodos no funcionan durante más de un día, contacte con AWS Support.

También puede definir una alarma de CloudWatch para recibir un aviso cuando se produzca este problema.

> **Note**
>
> La métrica Nodos totales no es precisa durante los cambios en la configuración del clúster y durante el mantenimiento rutinario del servicio. Este es el comportamiento esperado. La métrica informará en breve del número correcto de nodos del clúster. Para obtener más información, consulte the section called “Cambios de configuración”.

Para proteger los clústeres frente a terminaciones y reinicios inesperados de los nodos, cree al menos una réplica para cada índice en su dominio de OpenSearch Service.

Límite máximo de fragmentos superado

OpenSearch y las versiones 7.x de Elasticsearch tienen una configuración predeterminada de no más de 1000 fragmentos por nodo. OpenSearch/ElasticSearch genera un error si una solicitud, como crear un nuevo índice, provocara que se supere este límite. Si detecta este error, dispone de varias opciones:

- Agregue más nodos de datos al clúster.
- Aumente la configuración `_cluster/settings/cluster.max_shards_per_node`.
- Utilice la API _shrink para reducir el número de particiones en el nodo.

Dominio atascado en estado de procesamiento

El dominio de OpenSearch Service entra en el estado “Processing” (Procesando) cuando se encuentra en medio de un cambio de configuración. Si se inicia un cambio de configuración, el estado del dominio cambia a “Processing” (Procesando) mientras OpenSearch Service crea un nuevo entorno. En el nuevo entorno, OpenSearch Service lanza un nuevo conjunto de nodos aplicables (tales como datos, maestro o UltraWarm). Una vez que finaliza la migración, se terminan los nodos más antiguos.

El clúster puede quedarse atascado en el estado “Processing” (Procesando) si se produce una de estas situaciones:

- No se lanza un nuevo conjunto de nodos de datos.
- La migración de particiones al nuevo conjunto de nodos de datos no se realiza correctamente.
- La comprobación de validación ha fallado con errores.

Para ver pasos de resolución detallados para cada una de estas situaciones, consulte Why is my Amazon OpenSearch Service domain stuck in the “Processing” state?.

Bajo balance de ráfaga EBS

OpenSearch Service le envía una notificación de consola cuando el balance de ráfaga de EBS en uno de sus volúmenes de uso general (SSD) está por debajo del 70 % y una notificación de seguimiento si el balance cae por debajo del 20 %. Para solucionar este problema, puede escalar verticalmente el clúster o reducir las IOPS de lectura y escritura para que se pueda acreditar el balance de ráfaga. El balance de ráfagas se mantiene en 0 para los dominios con tipos de volúmenes gp3, así como para los dominios con volúmenes gp2 que tengan un tamaño de volumen superior a 1000 GiB. Para obtener más información, consulte Volúmenes de SSD de uso general (gp2). Puede monitorizar el balance de ráfagas de EBS con la métrica `BurstBalance` de CloudWatch.

La métrica de EBS aumenta durante el redimensionamiento del volumen

Al modificar los tamaños de volumen de Amazon Elastic Block Store, es posible que observe aumentos temporales en varias métricas de EBS como `Write Throughput`, `Write Throughput Micro bursting`, `Disk Queue Depth` y `IOPS`. Este es el comportamiento esperado durante la operación de cambio de tamaño y, por lo general, dura unos segundos. Sin embargo, la duración puede variar en función del tamaño del volumen que se modifique.

Para evitar problemas de latencia y rechazos de solicitudes, realice cambios de tamaño del volumen de EBS solo cuando el clúster esté en buen estado y el tráfico de red sea bajo.

No se pueden habilitar los registros de auditoría

Cuando intente habilitar la publicación de registro de auditoría mediante la consola del servicio de OpenSearch, puede encontrar el error siguiente:

La política de acceso a recursos especificada para el grupo de registros de CloudWatch Logs no concede permisos suficientes para que Amazon OpenSearch Service cree una secuencia de registro. Compruebe la política de acceso a recursos.

Si detecta este error, compruebe que el `resource`elemento de su política incluye el ARN del grupo de registro correcto. Si lo hace, siga estos pasos:

1. Espere varios minutos.
2. Actualice la página en el navegador web.
3. Seleccione Seleccionar grupo existente.
4. Para Grupo de registros existente, seleccione el grupo de registro que ha creado antes de recibir el mensaje de error.
5. En la sección de políticas de acceso, seleccione Seleccionar política existente.
6. Para Política existente, elija la política que ha creado antes de recibir el mensaje de error.
7. Seleccione Habilitar.

Si el error persiste después de repetir el proceso varias veces, póngase en contacto con el servicio de Soporte de AWS.

No se puede cerrar el índice

OpenSearch Service es compatible con la API `_close` solo para las versiones 7.4 y posteriores de OpenSearch y Elasticsearch. Si está usando una versión anterior está restaurando un índice desde una instantánea, puede eliminar el índice existente (antes o después de volver a indexarlo).

Verificaciones de licencias del cliente

Las distribuciones predeterminadas de Logstash y Beats incluyen una comprobación de licencia de propiedad exclusiva y no se puede conectar a la versión de código abierto de OpenSearch. Asegúrese de utilizar las distribuciones de Apache 2.0 (OSS) de estos clientes con OpenSearch Service.

Limitación controlada de solicitudes

Si recibe errores persistentes de `403 Request throttled due to too many requests` o `429 Too Many Requests`, considere la posibilidad de escalar verticalmente. Amazon OpenSearch Service realiza una limitación controlada de las solicitudes si la carga puede provocar que el uso de memoria supere el tamaño máximo de la pila de Java.

No se puede usar SSH en nodo

No se puede utilizar SSH para obtener acceso a ninguno de los nodos del clúster de OpenSearch, y no puede modificar directamente `opensearch.yml`. En su lugar, utilice la consola, AWS CLI o los SDK para configurar su dominio. Puede especificar unos ajustes en el nivel de clúster mediante las API de OpenSearch REST. Para obtener más información, consulte la Referencia de API de Amazon OpenSearch Service y the section called “Operaciones admitidas”.

Si necesita más información sobre el rendimiento del clúster, puede publicar los registros de errores y los registros lentos en CloudWatch.

Error de instantánea "No válido para la clase de almacenamiento del objeto"

Las instantáneas de OpenSearch Service no admiten la clase de almacenamiento de Amazon Glacier. Es posible que aparezca este error al intentar mostrar las instantáneas si el bucket de S3

incluye una regla de ciclo de vida que traslada los objetos a la clase de almacenamiento de Amazon Glacier.

Si tiene que restaurar una instantánea desde el bucket, restaure los objetos de Amazon Glacier, copie los objetos a un nuevo bucket y registre el nuevo bucket como repositorio de instantáneas.

Encabezado de host no válido

OpenSearch Service requiere que los clientes especifiquen `Host` en los encabezados de solicitud. Un valor de `Host` válido es el punto de conexión del dominio sin `https://`, como:

```
Host: search-my-sample-domain-ih2lhn2ew2scurji.us-west-2.es.amazonaws.com
```

Si recibe un error de `Invalid Host Header` al realizar una reserva, compruebe que su cliente o proxy incluye el punto de enlace del dominio de OpenSearch Service (y no, por ejemplo, su dirección IP) en el encabezado `Host`.

Tipo de instancia M3 no válido

OpenSearch Service no admite agregar o modificar instancias M3 a dominios existentes que ejecutan OpenSearch o Elasticsearch versión 6.7 o posteriores. Puede seguir utilizando instancias M3 con Elasticsearch 6.5 y versiones anteriores.

Recomendamos elegir un tipo de instancia más reciente. Para dominios que ejecutan OpenSearch o Elasticsearch 6.7 o versiones posteriores, se aplica la siguiente restricción:

- Si el dominio existente no utiliza instancias M3, ya no podrá cambiar a ellas.
- Si cambia un dominio existente de un tipo de instancia M3 a otro tipo de instancia, no puede volver a cambiar.

Las consultas activas dejan de funcionar después de habilitar UltraWarm

Cuando habilita UltraWarm en un dominio, si no hay anulaciones preexistentes en la configuración de `search.max_buckets`, OpenSearch Service establece automáticamente el valor en `10000` para

evitar que las consultas con gran cantidad de memoria saturen los nodos calientes. Si sus consultas activas utilizan más de 10 000 buckets, es posible que dejen de funcionar cuando habilite UltraWarm.

Dado que no puede modificar esta configuración debido a la naturaleza administrada de Amazon OpenSearch Service, debe abrir un caso de soporte para aumentar el límite. Los aumentos de límite no requieren una suscripción de premium support.

No se puede cambiar a una versión anterior después de una actualización

Las actualizaciones in situ son irreversibles, pero si se pone en contacto con AWS Support, pueden ayudarle a restaurar la instantánea automática previamente actualizada en un nuevo dominio. Por ejemplo, si actualiza un dominio de Elasticsearch 5.6 a 6.4, AWS Support puede ayudarle a restaurar la instantánea previamente actualizada en un nuevo dominio de Elasticsearch 5.6. Si ha realizado una instantánea manual del dominio original, puede realizar ese paso usted mismo.

Resumen necesario de dominios para todas las Regiones de AWS

El siguiente script utiliza el comando de Amazon EC2 describe-regions de AWS CLI para crear una lista de todas las regiones en las que OpenSearch Service podría estar disponible. A continuación, llama a list-domain-names para cada región:

```
for region in `aws ec2 describe-regions --output text | cut -f4`
do
    echo "\nListing domains in region '$region':"
    aws opensearch list-domain-names --region $region --query 'DomainNames'
done
```

Recibe el siguiente resultado para cada región:

```
Listing domains in region:'us-west-2'...
[
  {
    "DomainName": "sample-domain"
  }
]
```

Las regiones en las que OpenSearch Service no está disponible devuelven el error "Could not connect to the endpoint URL" (No se pudo conectar con la URL del punto de enlace).

Error del navegador al usar paneles de OpenSearch

El navegador incluye los mensajes de error de servicio en los objetos de respuesta HTTP cuando usa Dashboards para ver datos de su dominio de OpenSearch Service. Puede usar las herramientas para desarrolladores que suele haber disponibles en los navegadores web, como el Modo desarrollador de Chrome, para ver los errores de servicio subyacentes y como ayuda en las tareas de depuración.

Para ver errores de servicio en Chrome

1. Desde la barra del menú superior de Chrome, seleccione Ver, Desarrollador, Herramientas de desarrollador.
2. Seleccione la pestaña Red.
3. En la columna Estado, seleccione cualquier sesión HTTP que tenga un estado de 500.

Para ver errores de servicio en Firefox

1. Desde el menú, seleccione Herramientas, Desarrollador de web, Red.
2. Seleccione cualquier sesión HTTP que tenga un estado de 500.
3. Seleccione la pestaña Respuesta para ver la respuesta de servicio.

Partición de nodos y sesgo de almacenamiento

El sesgo de partición de nodo es cuando uno o más nodos de un clúster tienen significativamente más particiones que los demás nodos. El sesgo de almacenamiento de nodo es cuando uno o más nodos de un clúster tienen mucho más almacenamiento (`disk.indices`) que los demás nodos. Si bien estas dos condiciones pueden ocurrir temporalmente, por ejemplo, cuando un dominio ha reemplazado un nodo y todavía le está asignando particiones, debe abordarlas si persisten.

Para identificar ambos tipos de sesgo, ejecute la operación de la API _cat/allocation y compare las entradas `shards` y `disk.indices` en la respuesta:

```
shards    | disk.indices  | disk.used  | disk.avail   | disk.total   | disk.percent |
host     | ip       | node
   264    |      465.3mb  |   229.9mb  |      1.4tb   |      1.5tb   |            0 |
 x.x.x.x  | x.x.x.x  | node1
```

```
   115    |        7.9mb   |    83.7mb   |     49.1gb    |     49.2gb    |           0 |
x.x.x.x   | x.x.x.x   | node2
   264    |      465.3mb   |   235.3mb   |      1.4tb    |      1.5tb    |           0 |
 x.x.x.x   | x.x.x.x   | node3
   116    |        7.9mb   |    82.8mb   |     49.1gb    |     49.2gb    |           0 |
x.x.x.x   | x.x.x.x   | node4
   115    |        8.4mb   |      85mb   |     49.1gb    |     49.2gb    |           0 |
x.x.x.x   | x.x.x.x   | node5
```

Si bien es normal que haya algún sesgo de almacenamiento, cualquier valor por encima del 10 % del promedio es significativo. Cuando la distribución de las particiones está sesgada, el uso de la CPU, la red y el ancho de banda del disco también puede verse sesgado. Debido a que una mayor cantidad de datos generalmente implica más operaciones de indexación y búsqueda, los nodos más pesados también tienden a ser los nodos con mayor carga de recursos, mientras que los nodos más livianos representan una capacidad infrautilizada.

Corrección: utilice recuentos de particiones que sean múltiplos del recuento de nodos de datos para garantizar que cada índice se distribuya de manera uniforme entre los nodos de datos.

Partición del índice y el sesgo de almacenamiento

El sesgo de partición del índice es cuando uno o más nodos contienen más particiones de un índice que los otros nodos. El sesgo de almacenamiento de índice es cuando uno o más nodos contienen una cantidad desproporcionadamente grande del almacenamiento total de un índice.

El sesgo del índice es más difícil de identificar que el sesgo del nodo porque requiere cierta manipulación de la salida de la API _cat/shards. Investigue el sesgo del índice si hay algún indicio de sesgo en las métricas de clúster o nodo. A continuación, se indican algunos indicios comunes del sesgo de índice:

- Errores HTTP 429 que se producen en un subconjunto de nodos de datos
- Colocación en cola de operaciones de búsqueda o índice desigual en los nodos de datos
- Utilización desigual de la CPU o la pila de JVM en los nodos de datos

Corrección: utilice recuentos de particiones que sean múltiplos del recuento de nodos de datos para garantizar que cada índice se distribuya de manera uniforme entre los nodos de datos. Si sigue viendo un sesgo de almacenamiento de índices o partición, es posible que tenga que forzar una reasignación de particiones, que se produce con cada implementación azul/verde de su dominio de OpenSearch Service.

Operación no autorizada tras seleccionar acceso a la VPC

Cuando crea un dominio a través de la consola de OpenSearch Service, tiene la opción de seleccionar el acceso público o el acceso a la VPC. Si selecciona Acceso a la VPC, OpenSearch Service consulta la información de la VPC y genera un error si no tiene permisos adecuados:

```
You are not authorized to perform this operation. (Service: AmazonEC2; Status Code:
 403; Error Code: UnauthorizedOperation
```

Para poder realizar esta consulta, debe tener acceso a las operaciones `ec2:DescribeVpcs`, `ec2:DescribeSubnets` y `ec2:DescribeSecurityGroups`. Este requisito solo es para la consola. Si utiliza la CLI de AWS para crear y configurar un dominio con un punto de conexión de la VPC, no es necesario que tenga acceso a dichas operaciones.

Bloqueo al cargar después de crear el dominio de la VPC

Después de crear un dominio que utiliza el acceso a la VPC, es posible que la operación Configuration state (estado de configuración) del dominio no pase del estado Loading (Cargando). Si se produce este problema, es probable que AWS Security Token Service (AWS STS) esté desactivado para su región.

Para añadir puntos de enlace de la VPC a la VPC, OpenSearch Service debe asumir el rol `AWSServiceRoleForAmazonOpenSearchService`. Por lo tanto, AWS STS debe estar habilitado para crear nuevos dominios que utilicen el acceso a la VPC en una región determinada. Para obtener más información sobre la activación y desactivación de AWS STS, consulte la Guía del usuario de IAM.

Solicitudes denegadas a la API de OpenSearch

Con la introducción del control de acceso basado en etiquetas para la API de OpenSearch, es posible que empiece a ver errores de acceso denegado donde no lo hacía antes. Esto puede deberse a que una o más de sus políticas de acceso contienen `Deny` y utilizan la condición `ResourceTag`, y ahora se están respetando esas condiciones.

Por ejemplo, la política siguiente solo denegaba el acceso a la `CreateDomain` de la API de configuración, si el dominio tenía la etiqueta `environment=production`. Aunque la lista de

acciones también incluye `ESHttpPut`, la declaración de denegación no se aplicó a esa acción ni a ninguna otra acción `ESHttp*`.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [{
    "Action": [
      "es:CreateDomain",
      "es:ESHttpPut"
    ],
    "Effect": "Deny",
    "Resource": "*",
    "Condition": {
      "ForAnyValue:StringEquals": {
        "aws:ResourceTag/environment": [
          "production"
        ]
      }
    }
  }]
}
```

Con la compatibilidad agregada de etiquetas para los métodos HTTP de OpenSearch, una política basada en la identidad de IAM como la anterior dará lugar a que se deniegue el acceso a la acción `ESHttpPut` al usuario adjunto. Anteriormente, en ausencia de validación de etiquetas, el usuario adjunto habría podido enviar solicitudes PUT.

Si empieza a ver errores de acceso denegado después de actualizar sus dominios al software de servicio R20220323 o posterior, compruebe las políticas de acceso basadas en identidad para ver si es así y actualícelas si es necesario para permitir el acceso.

No se puede conectar desde Alpine Linux

Alpine Linux limita el tamaño de respuesta de DNS a 512 bytes. Si intenta conectarse a su dominio de OpenSearch Service desde Alpine Linux versión 3.18.0 o anterior, la resolución de DNS puede fallar si el dominio está en una VPC y tiene más de 20 nodos. Si utiliza una versión de Alpine Linux

superior a la 3.18.0, debería poder resolver más de 20 hosts. Para obtener más información, consulte las notas de la versión de Alpine Linux 3.18.0.

Si su dominio está en una VPC, le recomendamos que utilice otras distribuciones de Linux, como Debian, Ubuntu, CentOS, Red Hat Enterprise Linux o Amazon Linux 2, para conectarse a él.

Hay demasiadas solicitudes de Search Backpressure

El control de admisión basado en la CPU es un mecanismo de control que limita de forma proactiva el número de solicitudes a un nodo en función de su capacidad actual, tanto en caso de incrementos orgánicos como de picos de tráfico. Las solicitudes excesivas devuelven un código de estado HTTP 429 “Demasiadas solicitudes” en caso de rechazo. Este error indica que los recursos del clúster son insuficientes, que las solicitudes de búsqueda consumen muchos recursos o que se ha producido un aumento imprevisto de la carga de trabajo.

Search Backpressure proporciona el motivo del rechazo, lo que puede ayudar a ajustar las solicitudes de búsqueda que consumen muchos recursos. En caso de picos de tráfico, recomendamos volver a intentarlo desde el lado del cliente, con un retroceso y una fluctuación exponenciales.

Error de certificado cuando se utiliza un SDK

Dado que los SDK de AWS utilizan certificados de entidad de certificación desde su equipo, los cambios en los certificados en los servidores de AWS pueden provocar errores de conexión cuando intentar usar un SDK. Los mensajes de error varían, pero suelen contener el siguiente texto:

```
Failed to query OpenSearch
...
SSL3_GET_SERVER_CERTIFICATE:certificate verify failed
```

Puede evitar estos errores manteniendo actualizados los certificados de CA de su equipo y el sistema operativo. Si detecta este problema en un entorno corporativo y no administra su propio equipo, es posible que deba pedir ayuda a un administrador con el proceso de actualización.

La siguiente lista muestra las versiones mínimas necesarias del sistema operativo y Java:

- Las versiones de Microsoft Windows con actualizaciones instaladas a partir de enero de 2005 o fechas posteriores, contienen al menos uno de los CA necesarios en su lista de confianza.

- Mac OS X 10.4 con Java para Mac OS X 10.4 Release 5 (febrero de 2007), Mac OS X 10.5 (octubre de 2007) y las versiones posteriores contienen al menos uno de los CA necesarios en su lista de confianza.
- Red Hat Enterprise Linux 5 (marzo de 2007), 6 y 7 y CentOS 5, 6 y 7 contienen al menos uno de los CA necesarios en su lista de confianza predeterminada.
- Java 1.4.2_12 (mayo de 2006), 5 Update 2 (marzo de 2005) y todas las versiones posteriores, incluido Java 6 (diciembre de 2006), 7 y 8, contienen al menos uno de los CA necesarios en su lista de confianza predeterminada.

Las tres entidades de certificación son:

- Amazon Root CA 1
- Starfield Services Root Certificate Authority - G2
- Starfield Class 2 Certification Authority

Los certificados raíz de las primeras dos entidades están disponibles en Amazon Trust Services, aunque mantener el equipo actualizado es la solución más sencilla. Para obtener más información sobre certificados proporcionados por ACM, consulte Preguntas frecuentes sobre AWS Certificate Manager.

> **Note**
>
> En la actualidad, los dominios de OpenSearch Service de la región us-east-1 utilizan certificados de una entidad diferente. Tenemos previsto actualizar la región para utilizar estas nuevas entidades de certificación en un futuro próximo.

La instalación del complemento personalizado falla debido a la compatibilidad de las versiones

Problema: la instalación del complemento falló debido a una discrepancia de versión entre el complemento y la instancia de OpenSearch en ejecución. La función devuelve el siguiente error:

```
PluginValidationFailureReason : The provided plugin could not be loaded.
```

Causa: el complemento se compiló para OpenSearch ${*MAJOR*}.${*MINOR*}.{*PATCH*}, pero su entorno ejecuta OpenSearch ${*MAJOR*}.${*MINOR*}.0. OpenSearch requiere una coincidencia exacta de versiones entre los complementos y la instalación principal de OpenSearch por motivos de estabilidad y seguridad.

Posible solución: cree el complemento con la versión de OpenSearch ${*MAJOR*}.${*MINOR*}.0 para que coincida con la versión de su clúster.

Para verificar y actualizar la versión de OpenSearch

1. Utilice la API o el panel de control del clúster para ejecutar el siguiente comando. Reemplace los *valores de marcador de posición predeterminados* con su propia información.

   Solicitud de API:

   ```
   curl -X GET your-opensearch-endpoint/
   ```

   Consola de Dev Tools en el panel de control:

   ```
   GET /
   ```

   El comando devuelve información similar a la siguiente.

   ```
   {
     "name": "node-id",
     "cluster_name": "account-id:domain-name",
     "cluster_uuid": "cluster-uuid",
     "version": {
       "distribution": "opensearch",
       "number": "2.17.0",
       "build_type": "tar",
       "build_hash": "unknown",
       "build_date": "2024-12-17T11:00:09.799828091Z",
       "build_snapshot": false,
       "lucene_version": "9.11.1",
       "minimum_wire_compatibility_version": "7.10.0",
       "minimum_index_compatibility_version": "7.0.0"
     },
     "tagline": "The OpenSearch Project: https://opensearch.org/"
   }
   ```

2. Si el número de versión no es ${*MAJOR*}.${*MINOR*}.0, reconstruya el complemento de la siguiente manera:

   a. Actualice el complemento `descriptor.properties` para especificar la versión ${*MAJOR*}.${*MINOR*}.0.

   b. Reconstruya el complemento con el comando correspondiente a su tipo de proyecto.

   c. Ejecute el comando update-package con el archivo recién creado `.zip`.

   d. Ejecute el comando associate-package para asociar la última versión del complemento que se creó al ejecutar el comando `update-package` en el paso anterior.

Historial de documentos de Amazon OpenSearch Service

En este tema se describen cambios importantes realizados en Amazon OpenSearch Service. Las actualizaciones de software de servicio agregan compatibilidad con nuevas características, parches de seguridad, correcciones de errores y otras mejoras. Para utilizar las nuevas características, tal vez deba actualizar el software de servicio de su dominio. Para obtener más información, consulte the section called “Actualizaciones del software de servicio”.

Las características de servicio se implementan de forma incremental en las Regiones de AWS donde un servicio está disponible. Actualizamos esta documentación solo para la primera versión. No proporcionamos información sobre la disponibilidad en regiones ni anunciamos lanzamientos posteriores en regiones. Para obtener información sobre la disponibilidad regional de las características de servicio y para suscribirse a las notificaciones sobre actualizaciones, consulte Novedades de AWS

Para recibir notificaciones sobre las actualizaciones, puede suscribirse a la fuente RSS.

> **Note**
>
> Versiones de revisión: las versiones de software de servicio que terminan en “-P” y un número, como R20211203-P4, son versiones de revisión. Es probable que las revisiones incluyan mejoras de rendimiento, correcciones de errores menores y correcciones de seguridad o mejoras de posición. Dado que las revisiones no incluyen nuevas características ni cambios extraordinarios, por lo general no tienen un impacto directo en el usuario ni la documentación, y por ese motivo no se incluyen los detalles específicos de cada revisión en este historial de documentos.

Cambio	Descripción	Fecha
Compatibilidad con familias de instancias basadas en Graviton4 (c8g, m8g, r8g, r8gd)	Amazon OpenSearch Service ahora admite familias de instancias basadas en Graviton4 (c8g, m8g, r8g, r8gd). Para obtener más información, consulte Tipos	17 de octubre de 2025

	de instancias compatibles en Amazon OpenSearch Service.	
Nueva característica: machine learning de OpenSearch Ingestion (inferencia en lotes offline	Las canalizaciones de Amazon OpenSearch Ingestion (OSI) admiten el procesamiento de inferencia en lotes fuera de línea con machine learning (ML) para enriquecer de manera eficiente grandes volúmenes de datos a bajo costo. Utilice la inferencia en lotes sin conexión siempre que tenga conjuntos de datos grandes que puedan procesarse de forma asíncrona. La inferencia en lotes sin conexión funciona con los modelos de Amazon Bedrock y SageMaker. Esta característica está disponible en todas las Regiones de AWS compatibles con OpenSearch Ingestion con dominios de OpenSearch Service 2.17 y versiones posteriores. Para obtener más información, consulte Uso de una canalización de ingesta de OpenSearch con inferencia en lotes sin conexión de machine learning.	2 de octubre de 2025

Nueva característica: plantillas de CloudFormation para configurar la inferencia remota para el resaltado semántico	El resaltado semántico es una característica de búsqueda avanzada que mejora la relevancia de los resultados al analizar el significado y el contexto de las consultas en lugar de basarse únicamente en las coincidencias exactas de las palabras clave. Esta función utiliza modelos de machine learning para evaluar la similitud semántica entre las consultas de búsqueda y el contenido del documento, identificando y resaltando las frases o pasajes más relevantes desde el punto de vista contextual de los documentos. Amazon OpenSearch Service ahora incluye una plantilla de CloudFormation que configura automáticamente los modelos de Amazon SageMaker para el resaltado semántico. Para obtener más información, consulte Inferencia remota para plantillas de resaltado semántico.	26 de septiembre de 2025

La política administrada AmazonOpenSearchIngestionServiceRolePolicy se actualizó	La actualización otorga a OpenSearch Ingestion permiso para modificar los puntos de conexión de VPC creados por OpenSearch para compartir canalizaciones entre las VPC.	19 de septiembre de 2025
Nueva característica: configure las canalizaciones de OpenSearch Ingestion para la ingesta entre cuentas	Para los orígenes basados en push, como HTTP y OTel, Amazon OpenSearch Ingestion le permite compartir canalizaciones a través de Cuentas de AWS desde una nube privada virtual (VPC) hacia un punto de conexión de canalización en otra VPC separada. Para obtener más información, consulte Configuración de las canalizaciones de OpenSearch Ingestion para la transferencia entre cuentas	19 de septiembre de 2025
Amazon OpenSearch sin servidor admite la búsqueda vectorial basada en disco	La búsqueda vectorial basada en disco reduce considerablemente los costos operativos de las cargas de trabajo de vector en entornos con poca memoria. Para obtener más información, consulte Búsqueda vectorial basada en discos.	18 de septiembre de 2025

Compatibilidad con OpenSearch 3.1	Amazon OpenSearch Service ahora admite la versión de OpenSearch 3.1. Para obtener información sobre la compatibilidad de las actualizaciones, consulte Rutas de actualización compatibles. Para obtener más información sobre los cambios en las nuevas versiones, consulte la documentación de OpenSearch 3.0 y 3.1. Para obtener información sobre las operaciones de OpenSearch 3.1, consulte la referencia de la API de REST de OpenSearch o la referencia de la API del complemento específico.	9 de septiembre de 2025
Amazon OpenSearch sin servidor admite los requisitos del Estándar Federal de Procesamiento de Información (FIPS) 140-2	La norma FIPS 140-2 es un estándar del Gobierno estadounidense y canadiense que especifica los requisitos de seguridad para los módulos criptográficos que protegen la información confidencial. Los puntos de conexión FIPS de Amazon OpenSearch sin servidor están disponibles en Regiones de AWS donde FIPS es compatible. Estos puntos de conexión utilizan TLS 1.2 o una versión posterior y algoritmos criptográficos validados FIPS para todas las comunicaciones.	5 de septiembre de 2025

Nueva característica: control de acceso detallado basado en atributos SAML	Amazon OpenSearch Service admite un control de acceso detallado con SAML para mapear dinámicamente los usuarios y grupos de su proveedor de identidad a los usuarios y roles de control de acceso detallados de OpenSearch. Puede asignar estas funciones a dominios específicos de OpenSearch y colecciones sin servidor y definir los permisos a nivel de índice y la seguridad a nivel de los documentos. Para obtener más información, consulte Configuración del control de acceso detallado basado en atributos de SAML.	8 de agosto de 2025

Tema nuevo: puntos finales y cuotas de la interfaz de usuario de OpenSearch	La interfaz de Amazon OpenSearch amplía su experiencia modernizada de análisis operacional a nuevas Regiones de AWS, incluidas Asia Pacífico (Hyderabad), Asia Pacífico (Osaka), Asia Pacífico (Seúl), Europa (Milán), Europa (Zúrich), Europa (España) y Oeste de EE. UU. (Norte de California), lo que le permite obtener información a partir de datos que abarcan dominios administrados y colecciones sin servidor desde un único punto de conexión. La interfaz de usuario de OpenSearch puede conectarse a los dominios de OpenSearch anteriores a la versión 1.3 y a las colecciones sin servidor de OpenSearch. Ahora está disponible en 22 Regiones de AWS.	7 de agosto de 2025

Nuevas características: enriquecimiento semántico automático	El enriquecimiento semántico automático simplifica la implementación y las capacidades de la búsqueda semántica en Amazon OpenSearch Service. La búsqueda semántica arroja resultados de consulta que incorporan no solo la coincidencia de palabras clave, sino también la intención y el significado contextual de la búsqueda del usuario. Para obtener más información, consulte Acerca del enriquecimiento semántico automático. El enriquecimiento semántico automático ofrece los siguientes beneficios: Implementación simplificada No necesita conocimiento experto en machine learning (ML) ni integraciones complejas. Proceso automatizado El enriquecimiento semántico se produce automáticamente durante la ingesta de datos. Relevancia de búsqueda mejorada	6 de agosto de 2025

	El enriquecimiento semántico mejora la calidad y la precisión contextual de los resultados de búsqueda.
Escalabilidad	El enriquecimiento semántico aplica la búsqueda semántica a grandes conjuntos de datos sin intervención manual.

Nueva característica: Machine Learning en Amazon OpenSearch sin servidor	OpenSearch sin servidor ahora admite las capacidades de Machine Learning para flujos de trabajo de IA, como RAG (Retrieval-Augmented Generation) y búsqueda semántica. Utilice conectores para acceder a los algoritmos de aprendizaje automático y a los modelos remotos alojados en plataformas de terceros o en Servicios de AWS. Para obtener más información, consulte Configurar el machine learning en Amazon OpenSearch sin servidor. Machine Learning on OpenSearch sin servidor incluye las siguientes funciones: Búsqueda neuronal y búsqueda híbrida Realice operaciones de búsqueda semántica que comprendan el contexto y el significado de la consulta para obtener resultados más relevantes. Combine la búsqueda semántica y por palabra clave con la búsqueda híbrida para mejorar la relevancia. Para obtener más informaci	6 de agosto de 2025

ón, consulte Configurar la búsqueda neuronal y la búsqueda híbrida en OpenSearch sin servidor.

Conectores y modelos

Conéctese a modelos alojados en plataformas de aprendizaje automático de terceros y Servicios de AWS. Configure los conectores con modelos y despliéguelos para ejecutar predicciones y flujos de trabajo de IA.

Flujos de trabajo

Automatice las tareas complejas de configuración y preprocesamiento del aprendizaje automático encadenando varias llamadas a la API. Los flujos de trabajo simplifican la creación de aplicaciones de IA en OpenSearch. Para obtener más información, consulte Configurar flujos de trabajo en OpenSearch sin servidor.

Nuevas características: instantáneas de OpenSearch sin servidor	Las instantáneas son una copias de seguridad puntuales de sus coleccion es sin Amazon OpenSearc h sin servidor que ofrecen funciones de recuperación ante desastres. OpenSearch sin servidor crea y administr a automáticamente instantán eas de sus colecciones, a fin de garantizar la continuidad empresarial y la protección de los datos. Ventajas principales • Copias de seguridad automáticas cada hora sin necesidad de configuración manual • Gastos generales de mantenimiento cero • Sin costos de almacenam iento adicionales • Recuperación rápida en caso de pérdida accidental de datos • Posibilidad de restaurar índices específicos a partir de una instantánea Para obtener más información, consulte Copia de seguridad de las colecciones mediante instantáneas.	30 de julio de 2025

OpenSearch Ingestion admite dos fuentes nuevas	OpenSearch Ingestion ahora admite las siguientes fuentes: • Amazon Aurora • Amazon RDS	17 de julio de 2025

Nuevas características: integración de OpenSearch Service con Amazon S3 Vectors	Esta integración ofrece las siguientes características para mejorar considerablemente la búsqueda vectorial en OpenSearch Service. • Amazon S3 Vectors ofrece el primer almacén de objetos en la nube con compatibilidad nativa para almacenar y consultar vectores. S3 Vectors proporciona un almacenamiento vectorial rentable, elástico y duradero que se puede consultar en función del significado y la similitud semánticos. Ofrece tiempos de respuesta a las consultas inferiores a un segundo y reduce los costos de carga, almacenamiento y consulta de vectores hasta un 90 %. • Amazon OpenSearch Service ofrece la posibilidad de utilizar Amazon S3 como motor vectorial para índices vectoriales. Esta característica le permite descargar datos vectoriales a Amazon S3 y, al mismo tiempo, mantener las capacidades de búsqueda vectorial en menos de un segundo a un bajo costo.	15 de julio de 2025

	Para obtener más información, consulte (Vista previa) Integración de OpenSearch Service con Amazon S3 Vectors.	
Tema nuevo: Integración de Atlassian con OpenSearch Ingestion	Ahora usted puede usar canalizaciones de Amazon OpenSearch Ingestion para procesar datos de los servicios Atlassian Jira y Confluence. Esta integración le permite crear una base de conocimiento unificada y con capacidad de búsqueda al sincronizar proyectos completos de Jira y espacios de Confluence en OpenSearch. La integración mantiene la relevancia en tiempo real mediante la supervisión continua y la sincronización automática de las actualizaciones, al tiempo que ofrece opciones de filtrado flexibles para proyectos, tipos de problemas y contenido específicos. Para obtener más información, consulte Uso de una canalización de OpenSearch Ingestion con servicios de Atlassian.	5 de junio de 2025

Administración personalizada de complementos con la AWS CLI	Ahora puede administrar complementos personalizados para Amazon OpenSearch Service mediante la AWS CLI. Esta característica le permite instalar, actualizar, proteger y desinstalar complementos personalizados mediante un flujo de trabajo de línea de comandos simplificado. También puede proteger sus complementos con claves de AWS KMS y administrar las actualizaciones de los complementos cuando los actualice a versiones más recientes de OpenSearch. Para obtener más informaci ón, consulte Administración de complementos personali zados mediante la integraci ón de AWS CLI y el paquete personalizado AWS KMS de Amazon OpenSearch Service.	27 de mayo de 2025

Compatibilidad con OpenSearch 2.19	Amazon OpenSearch Service ahora admite la versión de OpenSearch 2.19. Para obtener más información, consulte las notas de la versión de OpenSearch 2.19. Para obtener información sobre las operaciones de OpenSearch 2.19, consulte la referencia de la API de REST de OpenSearch o la referencia de la API del complemento específico.	30 de abril de 2025

Nuevo soporte para especificar las funciones de canalización en OpenSearch Ingestion	Al crear o actualizar una canalización en Amazon OpenSearch Ingestion, deja de tener un rol de canalización específico en una configuración de canalización en formato YAML. En su lugar, en la consola, debe especificar el rol en un nuevo campo de rol de Pipeline. Si utiliza la AWS CLI, utilizará un parámetro `--pipeline-role-arn` nuevo. Para obtener más información, consulte los temas siguientes: • Configuración de roles y usuarios en Amazon OpenSearch Ingestion • Creación de canalizaciones de Amazon OpenSearch Ingestion • Tutorial: incorporar datos a un dominio mediante Amazon OpenSearch Ingestion • Tutorial: incorporar datos en una colección mediante Amazon OpenSearch Ingestion • CreatePipeline • UpdatePipeline	22 de abril de 2025

Nuevo tema	El tema de resolución de problemas La instalación de complementos personalizados falla debido a la compatibilidad de versiones aborda los pasos que debe seguir cuando encuentre el error `PluginValidationFailureReason : The provided plugin could not be loaded` . Este error suele deberse a una discordancia de versiones entre el complemento y la instancia de OpenSearch en ejecución.	21 de abril de 2025
Nueva característica: compatibilidad de la interfaz de usuario de OpenSearch con la búsqueda entre clústeres	La interfaz de OpenSearch ahora es compatible con la búsqueda entre clústeres . Esto le permite utilizar OpenSearch UI en una sola Región de AWS para acceder a los clústeres de una región diferente. Esto se hace configurándolo como un clúster remoto conectado a un clúster de la misma región. Para obtener más información, consulte Acceso a datos entre regiones y cuentas con búsqueda entre clústeres.	16 de abril de 2025

Contenido revisado: “Uso de la interfaz de usuario de OpenSearch en Amazon OpenSearch Service"	Hemos revisado y ampliado el capítulo Uso de la interfaz de usuario de OpenSearch en Amazon OpenSearch Service, que incluye procedimientos ampliados, detalles de configuración adicionales y un nuevo historial de versiones del soporte de Amazon OpenSearch Service para OpenSearch UI.	15 de abril de 2025
Amazon Q Developer para Amazon OpenSearch Service	La integración de Amazon Q con Amazon OpenSearch Service ofrece las siguientes funciones generativas: • Genere visualizaciones utilizando un lenguaje natural • Ver resúmenes e información sobre alertas • Consulta los resúmenes de los resultados de las consultas generados por Amazon Q en la página Discover • Vea los detectores de anomalías recomendados • Acceda al chat de Amazon Q para preguntas relacionadas con OpenSearch	31 de marzo de 2025

La política administrada AmazonOpenSearchServiceRolePolicy se actualizó	La política permite a OpenSearch actualizar el alcance de acceso de cualquier aplicación de AWS IAM Identity Center que solo sea administrada por OpenSearch.	28 de marzo de 2025
Instancias OR2 y OM2 para Amazon OpenSearch Service	Amazon OpenSearch Service ahora es compatible con instancias OR2 y OM2. En las pruebas internas, OR2 mostró un rendimiento de indexación hasta un 26 % mejor que OR1 y un 70 % más que R7. El OM2 se mostró hasta un 15 % mejor que el OR1 y un 66 % más que el M7g.	25 de marzo de 2025
Consulta directa para Registros de CloudWatch y Security Lake	Amazon OpenSearch Service ahora es compatible con consultas directas para consultar datos en Registros de CloudWatch y Security Lake.	1 de diciembre de 2024
índices k-NN	A partir de la versión 2.17 de OpenSearch, puede mover los índices K-nn a los niveles UltraWarm y de almacenamiento en frío.	13 de noviembre de 2024

Compatibilidad con OpenSearch	Amazon OpenSearch Service ahora es compatible con la versión 2.17 de OpenSearch. Esta versión incluye todas las características que formaban parte de las versiones 2.16 y 2.17. Para obtener más información, consulte las notas de las versiones 2.16 y 2.17.	13 de noviembre de 2024
Compatibilidad con OpenSearch	Amazon OpenSearch Service ahora es compatible con la versión 2.15 de OpenSearch. Esta versión incluye todas las características que formaban parte de las versiones 2.14 y 2.15. Para obtener más información, consulte las notas de las versiones 2.14 y 2.15.	11 de octubre de 2024
Nuevo rol vinculado a servicio de	Amazon OpenSearch Service agrega un rol vinculado a servicios denominado `AWSServiceRoleForOpensearchIngestionSelfManagedVpce` , que permite a Amazon OpenSearch Ingestion enviar datos de métricas a Amazon CloudWatch para canalizaciones con puntos de conexión de VPC autogestionados.	12 de junio de 2024

Consultas directas	Amazon OpenSearch Service ahora admite la ejecución de consultas directamente sobre los datos almacenados en Amazon S3, sin necesidad de ingerir los datos en un índice de OpenSearch.	22 de mayo de 2024
Compatibilidad con OpenSearch	Amazon OpenSearch Service ahora es compatible con la versión 2.13 de OpenSearch. Esta versión incluye todas las características que formaban parte de las versiones 2.12 y 2.13. Para obtener más información, consulte las notas de las versiones 2.12 y 2.13.	21 de mayo de 2024
Compatibilidad de Amazon OpenSearch Ingestion con la versión 2.7 de Data Prepper 2.7	Amazon OpenSearch Ingestion agrega compatibilidad con la versión 2.7 de Data Prepper. Para obtener más información, consulte las notas de la versión 2.7.	4 de abril de 2024
Servicio de AWS Acceso privado del a las colecciones de OpenSearch sin servidor	Ahora puede conceder acceso específico a los Servicios de AWS, como Amazon Bedrock, a sus colecciones de OpenSearch sin servidor dentro de una política de acceso a la red.	28 de marzo de 2024

Actualizaciones locales de EBS	Ahora puede hacer algunos cambios de EBS en sus dominios sin una implement ación azul/verde en Amazon OpenSearch Service.	14 de febrero de 2024
Visibilidad del cambio de configuración	Ahora puede hacer un seguimiento de los cambios en la configuración del dominio en la consola de Amazon OpenSearch Service y mediante la API de configura ción.	6 de febrero de 2024
Disponibilidad general de las colecciones de búsquedas vectoriales	Las colecciones de búsquedas vectoriales de Amazon OpenSearch sin servidor ya están disponibles de forma general. Durante la fase de versión preliminar se realizaro n las siguientes mejoras notables: • Las colecciones de búsquedas vectoriales ahora admiten cargas de trabajo con miles de millones de vectores, cada uno con hasta 128 dimensiones. • OpenSearch Dashboards ahora admite colecciones de búsquedas vectoriales.	29 de noviembre de 2023
Instancias OR	Amazon OpenSearch Service ahora admite los tipos de instancias OR1.	29 de noviembre de 2023

Consultas directas con Amazon S3 (versión preliminar)	Las consultas directas proporcionan una solución totalmente administrada que permite que los datos transaccionales estén disponibles en Amazon OpenSearch Service en cuestión de segundos después de escribirlos en un bucket de Amazon S3.	29 de noviembre de 2023
10 Capacidad de 10 TiB para colecciones de series temporales	Amazon OpenSearch sin servidor admite hasta 10 TiB de datos de índice para colecciones de series temporales. Esta versión también admite una capacidad máxima permitida de 200 OCU para todos los tipos de colecciones y la posibilidad de deshabilitar las réplicas en espera al crear una colección.	29 de noviembre de 2023
Compatibilidad con OpenSearch	Amazon OpenSearch Service ahora es compatible con la versión 2.11 de OpenSearch. Esta versión incluye todas las características que formaban parte de las versiones 2.10 y 2.11. Para obtener más información, consulte las notas de las versiones 2.10 y 2.11.	17 de noviembre de 2023

Compatibilidad de Amazon OpenSearch Ingestion con la versión 2.6 de Data Prepper 2.6	Amazon OpenSearch Ingestion agrega compatibilidad con la versión 2.6 de Data Prepper. Para obtener más información, consulte las notas de la versión 2.6. Además, puede especificar Amazon DynamoDB como origen de la canalización. Para obtener más información, consulte Using an OpenSearch Ingestion pipeline with Amazon DynamoDB.	17 de noviembre de 2023
Compatibilidad de Amazon OpenSearch Ingestion con la versión 2.5 de Data Prepper 2.5	Amazon OpenSearch Ingestion agrega compatibilidad con la versión 2.5 de Data Prepper. Para obtener más información, consulte las notas de la versión 2.5. Además, ahora puede especificar un dominio de OpenSearch Service o una colección de OpenSearch sin servidor como origen de la canalización. Para obtener más información, consulte OpenSearch source plugin en la documentación de Data Prepper.	17 de noviembre de 2023

Plantilla de CloudFormation para inferencias remotas	Para facilitar la configuración de la inferencia remota para búsquedas semánticas, Amazon OpenSearch Service proporciona una plantilla de AWS CloudFormation en la consola que automatiza el proceso de aprovisionamiento del modelo.	7 de noviembre de 2023
Actualización de la política de roles vinculados a servicios	Añade los permisos necesarios para que la política `AmazonOpenSearchServiceRolePolicy` de roles vinculados a servicios asigne y desasigne direcciones IPv6. La política `AmazonElasticsearchServiceRolePolicy` de Elasticsearch en desuso también se ha actualizado para garantizar la compatibilidad con versiones anteriores.	26 de octubre de 2023

Políticas de ciclo de vida de Amazon OpenSearch sin servidor	Amazon OpenSearch sin servidor presenta políticas de ciclo de vida de indexación para agilizar la administración de la retención y eliminación de datos. Ahora puede usar las API o una interfaz de configuración en la consola para establecer políticas de retención de datos para las colecciones de series temporales, lo que elimina la necesidad de crear índices o scripts diarios para eliminar datos antiguos.	25 de octubre de 2023
Compatibilidad con instancias Im4gn	Amazon OpenSearch Service ahora es compatible con las instancias Im4gn. Las instancias iM4GN están optimizadas para cargas de trabajo que gestionan grandes conjuntos de datos y necesitan una alta densidad de almacenamiento por vCPU.	20 de octubre de 2023

Opciones administrativas	Amazon OpenSearch Service ahora ofrece varias opciones administrativas que ofrecen un control detallado si necesita solucionar problemas con su dominio. Estas opciones incluyen la posibilidad de reiniciar el proceso de OpenSearch en un nodo de datos y la posibilidad de reiniciar un nodo de datos.	17 de octubre de 2023
Complementos opcionales	Amazon OpenSearch Service agrega compatibilidad con cuatro nuevos complemen tos de análisis de idiomas: los complementos Nori (coreano), Sudachi (japonés) , Pinyin (chino) y STConvert Analysis (chino), así como el complemento Search Ranking de Amazon Personalize.	16 de octubre de 2023
Compatibilidad con OpenSearch	Amazon OpenSearch Service ahora es compatible con la versión 2.9 de OpenSearch. Esta versión incluye todas las características que formaban parte de las versiones 2.8 y 2.9. Para obtener más información, consulte las notas de las versiones 2.8 y 2.9.	2 de octubre de 2023

Conectores de ML	Amazon OpenSearch Service agrega compatibilidad con conectores de machine learning (ML). Los conectores facilitan el acceso a los modelos de ML alojados en otros Servicios de AWS o en plataformas de machine learning (ML) de terceros.	6 de septiembre de 2023
Amazon OpenSearch Ingestion agrega compatibilidad con la versión 2.4 de Data Prepper 2.4	Amazon OpenSearch Ingestion agrega compatibilidad con la versión 2.4 de Data Prepper. Para obtener más información, consulte las notas de la versión 2.4. Además, ahora puede especificar Amazon Managed Streaming for Apache Kafka (Amazon MSK) como origen de la canalización.	31 de agosto de 2023
6 Capacidad de 6 TiB para colecciones de series temporales	Amazon OpenSearch sin servidor admite hasta 6 TiB de datos de índice para colecciones de series temporales. Esta versión también admite una capacidad máxima permitida de 100 OCU para las colecciones de series temporales y de búsqueda.	15 de agosto de 2023

Colecciones de búsqueda vectorial	Amazon OpenSearch sin servidor añade la opción de crear una colección de búsquedas vectoriales, que puede utilizar para almacenar incrustaciones vectoriales a fin de potenciar las búsquedas semánticas y de similitud.	26 de julio de 2023
Compatibilidad con OpenSearch	Amazon OpenSearch Service ahora es compatible con OpenSearch versión 2.7. Esta versión incluye todas las características que formaban parte de las versiones 2.6 y 2.7. Para obtener más información, consulte las notas de las versiones 2.6 y 2.7.	10 de julio de 2023
Compatibilidad con Data Prepper	Amazon OpenSearch Ingestion agrega compatibilidad con la versión 2.3 de Data Prepper. Para obtener más información, consulte las notas de la versión 2.3. Además, ahora puede especificar Amazon Security Lake como origen de la canalización.	26 de junio de 2023

Multi-AZ con modo de espera	Amazon OpenSearch Service agrega la opción de implement ar un dominio en tres zonas de disponibilidad (AZ), cada una de las cuales contiene una copia completa de los datos y los nodos de una de estas AZ actúan como modo de espera. La opción de implementación Multi-AZ con modo de espera ofrece una disponibilidad del 99,99 % y un rendimien to uniforme en caso de que se produzca un fallo en la infraestructura.	3 de mayo de 2023
Nuevo rol vinculado a servicio de	Amazon OpenSearch Service agrega un rol vinculado a un servicio denominado `AWSServiceRoleForA mazonOpenSearchIng estionService` , que permite a Amazon OpenSearc h Ingestion enviar datos de métricas a Amazon CloudWatc h.	26 de abril de 2023

Amazon OpenSearch Ingestion	Amazon OpenSearch Ingestion es un recopilador de datos totalmente gestionado que proporciona datos de registro y trazas en tiempo real a los dominios de OpenSearch Service y a las colecciones de OpenSearch sin servidor. OpenSearch Ingestion elimina la necesidad de utilizar soluciones de terceros, como Logstash o Jaeger, para introducir datos en sus dominios y colecciones.	26 de abril de 2023
Compatibilidad con OpenSearch	Amazon OpenSearch Service ahora es compatible con OpenSearch versión 2.5. Esta versión incluye todas las características que formaban parte de las versiones 2.4 y 2.5. Para obtener más información, consulte las notas de las versiones 2.4 y 2.5.	13 de marzo de 2023

Ventanas de menor actividad para mantenimiento	Amazon OpenSearch Service añade ventanas de menor actividad, es decir, bloques de tiempo diarios de 10 horas con poco tráfico durante los cuales puede programar las actualizaciones del software de servicio y las optimizac iones de ajuste automático que requieren una implement ación azul/verde. Las actualiza ciones en ventanas de menor actividad ayudan a minimizar la carga de los nodos maestros dedicados de un clúster durante los períodos de mayor tráfico. En el caso de los dominios nuevos creados después del 16 de febrero, la ventana de menor actividad se configura automáticamente para que funcione entre las 22:00 y las 8:00, hora local. Para los dominios existentes, debe habilitar la ventana de forma explícita.	16 de febrero de 2023

Configurar la autenticación SAML durante la creación del dominio	Amazon OpenSearch Service ahora permite configurar la autenticación SAML durante la creación del dominio. Anteriormente, había que configurar las opciones de SAML una vez que ya se había creado el dominio.	1 de febrero de 2023
Reindexación remota para dominios de VPC	Amazon OpenSearch Service incorpora la opción de una conexión de punto de conexión de VPC entre dos dominios. Ahora se puede utilizar reindexación remota para copiar índices de un dominio de VPC a otro sin un proxy inverso. Los dominios de VPC deben ejecutar el software de servicio R20221114 o posterior para utilizar esta característica.	31 de enero de 2023

Disponibilidad general de Amazon OpenSearch sin servidor	Amazon OpenSearch sin servidor ya está disponible con carácter general. Durante la fase de versión prelimina r se realizaron las siguientes mejoras notables: • Ahora la capacidad se puede reducir verticalmente hasta el mínimo de OCU configuradas cuando se produce una disminución del tráfico en el punto de conexión de la colección. • El número máximo de OCU permitido tanto para la indexación como para la búsqueda se incrementó de 20 a 50. Cada OCU incluye suficiente almacenamiento efímero en caliente para 120 GiB de datos de índice. • Ahora puede configurar los ajustes de acceso a datos mientras crea coleccion es, en lugar de tener que configurarlos en un flujo de trabajo independiente.	25 de enero de 2023

Ejecución de prueba asíncrona	Amazon OpenSearch Service ahora admite la ejecución de prueba asíncrona, lo que permite realizar una comprobación de validación antes de efectuar un cambio de configuración, e indica si los cambios provocarán una implementación azul/verde.	19 de enero de 2023
Nuevo rol vinculado a servicio de	Amazon OpenSearch Service agrega un rol vinculado a un servicio denominado `AWSServiceRoleForAmazonOpenSearchServerless` , que permite a OpenSearch sin servidor enviar datos de métricas a Amazon CloudWatch.	29 de noviembre de 2022
Versión preliminar de Amazon OpenSearch sin servidor	Amazon OpenSearch sin servidor es una configuración bajo demanda, sin servidor y de escalado automático para Amazon OpenSearch Service. Elimina las complejidades operativas del aprovisionamiento, la configuración y el ajuste de los clústeres de OpenSearch.	29 de noviembre de 2022

Compatibilidad con OpenSearch	Amazon OpenSearch Service ahora es compatible con OpenSearch versión 2.3. Esta versión incluye todas las características que formaban parte de las versiones 2.0, 2.1 y 2.2. Para obtener más información, consulte las notas de las versiones 2.0, 2.1, 2.2 y 2.3. La versión 2.3 contiene un cambio importante. Para obtener más información, consulte Rutas de actualización admitidas.	15 de noviembre de 2022
Compatibilidad con el complemento de Notificaciones	Amazon OpenSearch Service admite ahora el complemento de Notificaciones, que ofrece una ubicación central para todas las notificaciones de los complementos de OpenSearch. A partir de la versión 2.0, los destinos de alerta quedaron obsoletos y se sustituyeron por canales de notificación.	15 de noviembre de 2022

Compatibilidad con Kibana	Los dominios de Amazon OpenSearch Service que ejecutan Elasticsearch 7.1 ahora son compatibles con la última versión de revisión de Kibana 7.1.1, que incorpora correcciones de errores y mejora la seguridad. Cuando actualice los dominios de la versión 7.1 al software del servicio R20221114, OpenSearch Service los actualizará de forma automátic a a esta versión de revisión.	15 de noviembre de 2022
Compatibilidad con Kibana	Los dominios de Amazon OpenSearch Service que ejecutan Elasticsearch 6.8 ahora son compatibles con la última versión de revisión de Kibana 6.8.13, que incorpora correcciones de errores y mejora la seguridad. Cuando actualice los dominios de la versión 6.8 al software del servicio R20221114, OpenSearch Service los actualizará de forma automátic a a esta versión de revisión.	15 de noviembre de 2022

Compatibilidad con Kibana	Los dominios de Amazon OpenSearch Service que ejecutan Elasticsearch 6.3 ahora son compatibles con la última versión de revisión de Kibana 6.3.2, que incorpora correcciones de errores y mejora la seguridad. Cuando actualice los dominios de la versión 6.3 al software del servicio R20221114, OpenSearch Service los actualizará de forma automática a esta versión de revisión.	15 de noviembre de 2022

AWS PrivateLink	Con los puntos de conexión de VPC administrados por Amazon OpenSearch Service, puede conectarse de forma directa a los dominios de VPC de OpenSearch Service mediante un punto de conexión de VPC de interfaz en lugar de conectars e por Internet. Solo se puede acceder a un punto de conexión de VPC administr ado por OpenSearch Service dentro de la VPC donde se aprovisiona el punto de conexión o desde cualquier VPC interconectada con la VPC donde se aprovisiona el punto de conexión, según lo permitan las tablas de enrutamiento y los grupos de seguridad. Su dominio de VPC debe ejecutar el software de servicio R20220928 o posterior para conectarse a un punto de conexión de VPC de interfaz.	7 de noviembre de 2022
Correcciones de errores y mejoras de rendimiento	El software de servicio R20220928 incluye correccio nes de errores y mejoras de rendimiento, incluido un registro SAML mejorado. La actualización también cambia el inquilino predeterminado a `Global` en lugar de `Private`.	3 de octubre de 2022

Referencia de la API mejorada	Amazon OpenSearch Service ofrece una referencia de API de configuración integral y mejorada. Las nuevas referencias contienen todos los tipos de datos y acciones disponibles, ejemplos de sintaxis de solicitudes y respuestas, y enlaces a las referencias de SDK correspon dientes para todos los idiomas admitidos.	13 de septiembre de 2022
Validación azul-verde	Amazon OpenSearch Service ahora comprueba la validación antes de las implementaciones azul-verde y muestra errores de validación si su dominio no cumple con los requisitos para una actualización.	16 de agosto de 2022
Compatibilidad con OpenSearch	Amazon OpenSearch Service ahora es compatible con OpenSearch versión 1.3. Para obtener más informaci ón, consulte las notas de la versión 1.3.	27 de julio de 2022

Compatibilidad con el complemento ML Commons	Amazon OpenSearch Service agrega compatibilidad con el complemento ML Commons, que proporciona un conjunto de algoritmos de machine learning comunes a través del transporte y las llamadas de API de REST. También puede interactuar con el complemento ML Commons a través de comandos de PPL.	27 de julio de 2022
Compatibilidad con los volúmenes gp	Amazon OpenSearch Service incorpora compatibilidad con el tipo de volumen SSD de uso general de EBS gp3. Puede especificar las IOPS aprovisionadas y el rendimiento adicionales al crear o modificar el dominio.	26 de julio de 2022
Documentación de prácticas recomendadas mejoradas	La documentación de Amazon OpenSearch Service proporciona prácticas recomendadas operativas mejoradas y recomendaciones generales para crear y utilizar dominios de OpenSearch Service.	6 de julio de 2022
Integración con Service Quotas	Ahora puede ver las cuotas de Amazon OpenSearch Service y solicitar aumentos de cuotas desde la consola de Service Quotas.	29 de junio de 2022

Control de acceso basado en etiquetas para la API de OpenSearch	Ahora puede utilizar etiquetas para controlar el acceso a las API de OpenSearch. Anteriormente, solo podía utilizar etiquetas para controlar el acceso a la API de configuración.	16 de junio de 2022
Búsqueda entre clústeres en regiones	La búsqueda entre clústeres ahora se admite en Regiones de AWS siempre y cuando ambos dominios ejecuten Elasticsearch versión 7.10 o posterior, o cualquier versión de OpenSearch.	14 de junio de 2022
Compatibilidad con Kibana 5.6 individual	Amazon OpenSearch Service incorpora compatibilidad con Kibana 5.6.16 individual. Con Kibana 5.6.16 individual, puede utilizar Kibana 5.6 como front-end mientras se conecta a las versiones 5.1, 5.3, 5.5 y 5.6 de Elasticsearch. Debe estar en el software del servicio R20220323 o posterior para utilizar Kibana 5.6 individual.	4 de abril de 2022

R20220323-P1	Amazon OpenSearch Service lanzó recientemente la actualización del software del servicio R20220323, pero la actualización se revirtió posteriormente debido a un problema. Se recomiend a actualizar los dominios a la versión de revisión R20220323-P1 o posterior, que soluciona el problema.	4 de abril de 2022
Compatibilidad con OpenSearch	Amazon OpenSearch Service ahora es compatible con OpenSearch versión 1.2. Para obtener más informaci ón, consulte las notas de la versión 1.2.	4 de abril de 2022
Observabilidad	La instalación predeterminada de OpenSearch Dashboard s para Amazon OpenSearch Service incluye el complemen to Observabilidad, que se puede utilizar para visualiza r eventos controlados por datos mediante el lenguaje de procesamiento con canalizaciones (PPL) para explorar y consultar los datos. El complemento requiere OpenSearch 1.2 o posterior y el software del servicio R20220323 o posterior.	4 de abril de 2022

Compatibilidad con Kibana	Los dominios de Amazon OpenSearch Service que ejecutan Elasticsearch 7.7 ahora son compatibles con la última versión de revisión de Kibana 7.7, que incorpora correcciones de errores y mejora la seguridad. Cuando actualice los dominios 7.7 al software del servicio R20220323 o posterior, OpenSearch Service los actualizará automáticamente a esta versión de revisión.	4 de abril de 2022

Cambios en la métrica de presión de memoria de JVM	Amazon OpenSearch Service cambió la lógica de la métrica `JVMMemoryPressure` de CloudWatch para reflejar con mayor precisión la utilización de la memoria. Anteriormente, las métricas solo tenían en cuenta el grupo de memoria de antigua generación del montón de JVM. Con este cambio, la métrica también tiene en cuenta el grupo de memoria de nueva generación. Después de actualizar el dominio al software del servicio R20220323, es posible que observe un incremento en las métricas `JVMMemoryPressure`, `MasterJVMMemoryPressure` o `WarmJVMMemoryPressure`.	4 de abril de 2022
Diccionarios personalizados con el complemento IK (Chinese) Analysis	Amazon OpenSearch Service ahora admite el uso de diccionarios personalizados con el complemento IK (Chinese) Analysis.	4 de abril de 2022

Replicación entre clústeres en dominios existentes	Amazon OpenSearch Service eliminó la limitación de que solo se puede implementar la búsqueda en clústeres y la replicación entre clústeres en dominios creados a partir del 3 de junio de 2020 incluido. Ahora puede habilitar estas características en todos los dominios, independientemente de cuándo se hayan creado. Ambos dominios deben estar en el software del servicio R20220323 o posterior.	4 de abril de 2022
Visibilidad de implementación azul-verde	Amazon OpenSearch Service ofrece ahora más visibilidad del progreso de las implement aciones azul-verde. Puede supervisar estos detalles en la consola o mediante la API de configuración.	27 de enero de 2022

Control de acceso detallado en dominios existentes	Ahora puede habilitar el control de acceso detallado en los dominios existentes. Puede habilitar un período de migración temporal para las políticas de acceso abiertas o basadas en IP para garantizar que los usuarios puedan seguir accediendo a su dominio mientras crea y asigna roles. La habilitación del control de acceso detallado en los dominios existentes requiere el software de servicio R20211203 o posterior.	6 de enero de 2022
Roles de OpenSearch Dashboards que han cambiado de nombre	Con el software de servicio R20211203, se le ha cambiado el nombre al rol `kibana_user` por `opensearch_dashboards_user`, y a `kibana_read_only` se le ha cambiado el nombre por `opensearch_dashboards_read_only`. Este cambio es de aplicación a todos los dominios recién creados de OpenSearch 1.x. En los dominios de OpenSearch existentes que se actualice al software de servicio R20211203, los roles permanecen igual.	4 de enero de 2022

Soporte de OpenSearch	Amazon OpenSearch Service ahora soporta la versión de OpenSearch 1.1. Para obtener más información, consulte las notas de la versión 1.1.	4 de enero de 2022
Editor visual de ISM	La instalación predeterminada de OpenSearch Dashboard s para Amazon OpenSearc h Service ahora admite el editor visual para las políticas de ISM. Esta característica requiere OpenSearch 1.1 o posterior.	4 de enero de 2022
Actualización de la prevenció n del suplente confuso entre servicios	Amazon OpenSearch Service soporta el uso de las claves de contexto de condición global `aws:SourceArn` y `aws:SourceAccount` en las políticas de recursos de IAM para evitar el problema del suplente confuso. Debe estar en el software de servicio R20211203 o posterior para utilizar estas claves de condición.	4 de enero de 2022

Parche de Log4j	El software de servicio R20211203-P2 actualiza la versión de Log4j utilizada en OpenSearch Service según lo recomendado por los asesores de CVE-2021-44228 y CVE-2021-45046. El parche se aplica a los dominios que ejecutan todas las versiones de OpenSearch y Elasticsearch. OpenSearch Service continuará actualizando varias versiones de Log4j internamente y no estarán necesariamente restringidas a la última versión de Log4j. La versión de Log4j de su dominio depende de la versión de software que ejecute el dominio. Sin embargo, independientemente de la versión de Log4j, siempre que ejecute R20211203-P2 o posterior, sus dominios contienen la actualización de Log4j necesaria para abordar CVE-2021-44228 y CVE-2021-45046.	15 de diciembre de 2021

Replicación entre clústeres	La replicación entre clústeres permite replicar índices, asignaciones y metadatos de un dominio de OpenSearch Service a otro. La replicación entre clústeres requiere un dominio que ejecute Elasticsearch 7.10 o OpenSearch 1.1 o posterior.	5 de octubre de 2021
Nuevas políticas administradas de AWS	El lanzamiento de Amazon OpenSearch Service incluye nuevas políticas administradas de AWS y la obsolescencia de las políticas antiguas.	8 de septiembre de 2021
Compatibilidad con Kibana	Los dominios de Amazon OpenSearch Service que ejecutan la versión 6.4 de Elasticsearch heredada ahora son compatibles con la última versión 6.4 del parche para Kibana, que agrega correcciones de errores y mejora la seguridad. OpenSearch Service actualizará de forma automática los dominios a esta versión de parche.	8 de septiembre de 2021
Flujos de datos	Amazon OpenSearch Service agrega compatibilidad a flujos de datos, que simplifican el proceso de administración de datos de series temporales. Su dominio debe ejecutar OpenSearch 1.0 o posterior para utilizar flujos de datos.	8 de septiembre de 2021

Amazon OpenSearch Service	AWS cambia el nombre de Amazon OpenSearch Service para eliminar la marca heredada “Elasticsearch”. Amazon OpenSearch Service admite OpenSearch y Elasticsearch OSS heredado. Al crear un clúster, tiene la opción de elegir qué motor de búsqueda utilizar. OpenSearch Service ofrece una amplia compatibilidad con Elasticsearch OSS 7.10, la versión final de código abierto del software.	8 de septiembre de 2021
Almacenamiento en frío	El almacenamiento en frío es un nuevo nivel de almacenamiento para datos históricos o a los que se accede con poca frecuencia. Los índices almacenados en frío solo ocupan el almacenamiento de S3 y no tienen informática vinculada a ellos. El almacenamiento en frío requiere un dominio que ejecute Elasticsearch 7.9 o posterior y el software de servicio R20210426 o posterior.	13 de mayo de 2021

Instancias de Graviton basadas en ARM	Ahora, Amazon OpenSearch Service admite tipos de instancias de Graviton basadas en ARM (M6G, C6G, R6G y R6GD). Los tipos de instancias de Graviton están disponibles en dominios nuevos y existentes que ejecutan Elasticsearch 7.9 o posterior y el software de servicio R20210331 o posterior.	4 de mayo de 2021
Plantillas de ISM	Amazon OpenSearch Service agrega compatibilidad para las plantillas de ISM, que permiten adjuntar de forma automática una política de ISM a un índice, si este coincide con un patrón definido en la política. Las plantillas de ISM requieren el software de servicio R20210426 o posterior. Esta actualización también deja obsoleta la configuración `policy_id` , que significa que ya no puede utilizar plantillas de índice para aplicar políticas de ISM a los índices recién creados. La actualización presenta un cambio sustancial para las plantillas existentes de CloudFormation mediante esta configuración.	27 de abril de 2021

Compatibilidad con Elasticsearch	Amazon OpenSearch Service ahora es compatible con la versión 7.10 de Elasticsearch. Para obtener más información, consulte las notas de la versión de 7.10.	21 de abril de 2021
Búsqueda asíncrona	Ahora, Amazon OpenSearch Service admite la búsqueda asíncrona, que permite ejecutar solicitudes de búsqueda en segundo plano. La búsqueda asíncrona requiere un dominio que ejecute Elasticsearch 7.10 o posterior, y el software de servicio R20210331 o posterior.	21 de abril de 2021
Control de acceso basado en etiquetas para la API de configuración	Ahora puede utilizar etiquetas de AWS para controlar el acceso a la API de configuración de Amazon ES.	2 de marzo de 2021
Ajuste automático	Amazon OpenSearch Service agrega un Ajuste automático, que utiliza métricas de rendimiento y uso desde el clúster para sugerir cambios en la configuración de la JVM en sus nodos. El ajuste automático requiere un dominio que ejecute Elasticsearch 6.7 o posterior y el software de servicio R20201117 o posterior.	24 de febrero de 2021

Trace Analytics	La instalación predeterminada de Kibana para Amazon OpenSearch Service ahora incluye el complemento de Análisis de seguimiento, que permite monitorear los datos de seguimiento de sus aplicaciones distribuidas. El complemento requiere un dominio que ejecute Elasticsearch 7.9 o posterior y un software de servicio R20210201 o posterior.	17 de febrero de 2021
Métricas de particiones	Amazon OpenSearch Service agrega las siguientes métricas de CloudWatch para el seguimiento del estado de las particiones: `Shards.active`, `Shards.unassigned`, `Shards.delayedUnassigned` `Shards.activePrimary`, `Shards.initializing`, `Shards.relocating`. Las métricas están disponibles en dominios con el software de servicio R20210201 o posterior.	17 de febrero de 2021

Informes de Kibana	La instalación predeterminada de Kibana para Amazon OpenSearch Service ahora admite informes bajo demanda para las páginas Discover, Visualize y Dashboard. Esta característica requiere Elasticsearch 7.9 o posterior y el software de servicio R20210201 o posterior.	17 de febrero de 2021
Compatibilidad con Kibana	Los dominios de Amazon OpenSearch Service que ejecutan Elasticsearch 5.6 ahora son compatibles con la última versión de revisión de Kibana 5.6, que incorpora correcciones de errores y mejora la seguridad. Amazon ES actualizará de forma automática los dominios a esta versión de parche.	17 de febrero de 2021
Cifrado para dominios existentes	Amazon OpenSearch Service ahora admite la habilitación del cifrado de datos en reposo y el cifrado de nodo a nodo en dominios existentes que ejecutan Elasticsearch 6.7 o una versión posterior. Después de habilitar esta configuración, no podrá deshabilitarla.	27 de enero de 2021

Reindexación remota	Ahora, Amazon OpenSearch Service admite la reindexación remota, que permite migrar índices desde dominios remotos. Esta característica requiere el software de servicio R20201117 o una versión posterior.	24 de noviembre de 2020
Lenguaje de procesamiento con plecas	Ahora, Amazon OpenSearch Service admite el lenguaje de procesamiento de canalizaciones (PPL), un lenguaje de consulta que permite utilizar la sintaxis de canalización (|) para consultar datos almacenados en Elasticsearch. Esta característica requiere el software de servicio R20201117 o una versión posterior. Para obtener más información, consulte .	24 de noviembre de 2020
Cuadernos de Kibana	Amazon OpenSearch Service agrega compatibilidad con cuaderno de Kibana, que permite combinar visualizaciones en vivo y texto narrativo en una única interfaz. Esta característica requiere el software de servicio R20201117 o una versión posterior.	24 de noviembre de 2020

Diagramas de Gantt	La instalación predeterminada de Kibana para Amazon OpenSearch Service ahora admite un nuevo tipo de visualización: diagramas de Gantt. Esta característica requiere el software de servicio R20201117 o una versión posterior.	24 de noviembre de 2020
Compatibilidad con Elasticsearch	Amazon OpenSearch Service ahora es compatible con la versión 7.9 de Elasticsearch. Para obtener más información, consulte las notas de la versión de 7.9.	24 de noviembre de 2020
Actualizaciones de detección de anomalías	La detección de anomalías de Amazon OpenSearch Service agrega compatibilidad con la alta cardinalidad, que permite categorizar anomalías con una dimensión como dirección IP, ID del producto y código del país, entre otros. Esta característica requiere el software de servicio R20201117 o una versión posterior.	24 de noviembre de 2020

Actualizaciones de diccionario dinámicas	Ahora, Amazon OpenSearch Service permite actualizar los analizadores de búsqueda sin reindexar. Puede actualizar los archivos de diccionario en algunos o todos sus dominios y Amazon ES realiza un seguimiento de las versiones de los paquetes a lo largo del tiempo para que tenga un historial de lo que cambió y cuándo. Esta caracterí stica requiere el software de servicio R20201019 o una versión posterior.	17 de noviembre de 2020
Puntos de conexión personali zados	Ahora, Amazon OpenSearc h Service admite puntos de conexión personalizados, que le permiten dar a su dominio de Amazon ES una nueva URL. Si alguna vez intercamb ia dominios, puede mantener la misma URL. Esta caracterí stica requiere el software de servicio R20201019 o una versión posterior.	5 de noviembre de 2020
Nuevos complementos de idioma	Amazon OpenSearch Service ahora admite complemen tos de Análisis IK (chino), vietnamita y tailandés en dominios que ejecutan Elasticsearch 7.7 o posterior con el software de servicio R20201019 o posterior.	28 de octubre de 2020

Compatibilidad con Elasticsearch	Amazon OpenSearch Service ahora es compatible con la versión 7.8 de Elasticsearch. Para obtener más información, consulte las notas de la versión de 7.8.	28 de octubre de 2020
Autenticación SAML para Kibana	Ahora, Amazon OpenSearch Service admite la autenticación SAML para Kibana, que permite utilizar proveedores de identidad de terceros para iniciar sesión en Kibana, administrar el control de acceso detallado, buscar sus datos y crear visualizaciones. Esta característica requiere el software de servicio R20201019 o una versión posterior.	27 de octubre de 2020
Instancias T	Amazon OpenSearch Service ahora es compatible con las instancias `t3.small` y `t3.medium` .	23 de septiembre de 2020

Registros de auditoría	Amazon OpenSearch Service ahora es compatible con los registros de auditoría para sus datos, que permite realizar un seguimiento de los intentos fallidos de inicio de sesión, el acceso de los usuarios a los índices, los documentos, los campos y mucho más. Esta caracterí stica requiere el software de servicio R20200910 o una versión posterior.	16 de septiembre de 2020
Actualizaciones de UltraWarm	UltraWarm para Amazon OpenSearch Service agrega nuevas métricas, nueva configuración, una cola de migración más amplia y una API de cancelación. Estas actualizaciones requieren el software de servicio R20200910 o posterior. Para obtener más información, consulte .	14 de septiembre de 2020

Learning to Rank	Amazon OpenSearch Service admite ahora el complemento de Aprender a clasificar de código abierto, que permite utilizar tecnologías de machine learning para mejorar la relevancia de la búsqueda. Esta característica requiere el software de servicio R20200721 o una versión posterior.	27 de julio de 2020
Similitud coseno de k-NN	k-Nearest Neighbor (k-NN) ahora permite buscar “vecinos más cercanos” por similitud coseno además de la distancia euclidiana. Esta característica requiere el software de servicio R20200721 o una versión posterior.	23 de julio de 2020
Compresión con gzip	Amazon OpenSearch Service ahora es compatible con la compresión gzip para la mayoría de las solicitudes y respuestas HTTP, que puede reducir la latencia y conservar el ancho de banda. Esta característica requiere el software de servicio R20200721 o una versión posterior.	23 de julio de 2020

Compatibilidad con Elasticsearch	Amazon OpenSearch Service ahora es compatible con la versión 7.7 de Elasticsearch. Para obtener más información, consulte las notas de la versión de 7.7	23 de julio de 2020
Servicio de mapas de Kibana	La instalación predeterminada de Kibana para Amazon OpenSearch Service incluye un servidor de mapas WMS, excepto para los dominios de las regiones de India y China.	18 de junio de 2020
Mejoras de SQL	La compatibilidad de SQL para Amazon OpenSearch Service ahora admite muchas operaciones nuevas, una interfaz de usuario de Kibana dedicada a la exploración de datos y una CLI interactiva.	3 de junio de 2020
Búsqueda en clústeres	Amazon OpenSearch Service permite realizar consultas y agregaciones entre clústeres en varios dominios conectados.	3 de junio de 2020
Detección de anomalías	Amazon OpenSearch Service permite la detección de forma automática de anomalías casi en tiempo real.	3 de junio de 2020

UltraWarm	El almacenamiento UltraWarm para Amazon OpenSearch Service culminó la fase de versión preliminar pública y ahora está disponible con carácter general. La caracterí stica ahora admite un conjunto más amplio de versiones y Regiones de AWS. Para obtener más información, consulte .	5 de mayo de 2020
Diccionarios personalizados	Amazon OpenSearch Service permite cargar archivos de diccionario personalizados para utilizarlos con el clúster. Estos archivos mejoran los resultados de la búsqueda al indicarle a Elasticsearch que ignore ciertas palabras de alta frecuencia o que trate los términos como equivalentes.	21 de abril de 2020
Compatibilidad con Elasticsearch	Amazon OpenSearch Service ahora es compatible con la versión 7.4 de Elasticsearch. Para obtener más información, consulte Versiones compatibles.	12 de marzo de 2020
k-NN	Amazon OpenSearch Service agrega compatibilidad con la búsqueda k-Nearest Neighbor (k-NN). k-NN requiere el software de servicio R20200302 o una versión posterior.	3 de marzo de 2020

Index State Management	Amazon OpenSearch Service agrega la administración de estados de índice (ISM), que permite la automatiz ación de tareas rutinaria s, como eliminar índices cuando alcanzan cierta antigüedad. Esta caracterí stica requiere el software de servicio R20200302 o una versión posterior.	3 de marzo de 2020
Compatibilidad con Elasticse arch	Amazon OpenSearch Service ahora admite la última versión de revisión para la versión 5.6, que agrega correcciones de errores y mejora la seguridad . Amazon ES actualizará de forma automática los dominios 5.6 existentes a esta versión. Tenga en cuenta que esta versión de Elasticsearch informa incorrectamente su versión como 5.6.17.	2 de marzo de 2020
Control de acceso detallado	Amazon OpenSearch Service ahora admite el control de acceso detallado, que ofrece seguridad a nivel del índice, del documento y del campo, tenencias múltiples en Kibana y autenticación básica de HTTP opcional para el clúster.	11 de febrero de 2020

Almacenamiento UltraWarm (versión preliminar)	Amazon OpenSearch Service agrega UltraWarm, un nuevo nivel de almacenamiento templado que utiliza Amazon S3 y una sofisticada solución de almacenamiento en caché para mejorar el rendimiento. Para los índices en los que no se está escribiendo activamente y que se consultan con menos frecuencia, el almacenamiento UltraWarm ofrece costos significativamente más bajos por GiB.	3 de diciembre de 2019
Características de cifrado para las regiones de China	El cifrado de datos en reposo y el cifrado de nodo a nodo ya están disponibles en las regiones de China (Pekín) `cn-north-1` y China (Ningxia) `cn-northwest-1` .	20 de noviembre de 2019
Exigir HTTPS	Ahora puede exigir que todo el tráfico a sus dominios de Amazon ES llegue a través de HTTPS. Al configurar su dominio, marque la casilla Exigir HTTPS. Esta característica requiere el software de servicio R20190808 o una versión posterior.	3 de octubre de 2019

Compatibilidad con Elasticsearch 7.1 y	Amazon OpenSearch Service ahora es compatible con las versiones 7.1 y 6.8 de Elasticsearch. Para obtener más información, consulte Versiones compatibles.	13 de agosto de 2019
Instantáneas cada hora	En lugar de instantáneas diarias, Amazon OpenSearch Service ahora toma instantáneas cada hora de los dominios que ejecutan Elasticsearch 5.3 y versiones posteriores, para tener copias de seguridad más frecuentes desde las que podrá restaurar los datos.	8 de julio de 2019
Compatibilidad con Elasticsearch	Amazon OpenSearch Service ahora es compatible con la versión 6.7 de Elasticsearch. Para obtener más información, consulte Versiones compatibles.	29 de mayo de 2019
Compatibilidad con SQL	Ahora, Amazon OpenSearch Service permite la consulta de datos con SQL. La compatibilidad con SQL requiere el software de servicio R20190418 o una versión posterior.	15 de mayo de 2019

5-Tipos de instancias de la serie	Amazon OpenSearch Service ahora admite los tipos de instancias M5, C5 y R5. En comparación con los tipos de instancias de la generación anterior, estos nuevos tipos ofrecen un mejor rendimien to a precios más bajos. Para obtener más información, consulte Límites.	24 de abril de 2019
Compatibilidad con Elasticse arch	Amazon OpenSearch Service ahora es compatible con la versión 6.5 de Elasticsearch.	8 de abril de 2019
Alertas	Las alertas para Amazon OpenSearch Service notifican cuando los datos de uno o más índices de Amazon ES cumplen determinadas condiciones. Las alertas requieren el software de servicio R20190221 o una versión posterior.	25 de marzo de 2019
Compatibilidad con tres zonas de disponibilidad	Amazon OpenSearch Service ahora admite tres zonas de disponibilidad en varias regiones. Esta versión incluye también una experiencia de consola simplificada. Este multi-AZ requiere el software de servicio R20181023 o una versión posterior.	7 de febrero de 2019

Compatibilidad con Elasticsearch	Amazon OpenSearch Service ahora es compatible con la versión 6.4 de Elasticsearch.	23 de enero de 2019
200-Clústeres de 200 nodos	Ahora Amazon ES permite crear clústeres con hasta 200 nodos de datos y un total de 3 PB de almacenamiento.	22 de enero de 2019
Actualizaciones del software del servicio	Amazon ES ahora permite actualizar manualmente el software de servicio de su dominio con el fin de obtener beneficios de nuevas caracterí sticas con mayor rapidez o realizar las actualizaciones en el momento en que hay menos tráfico. Para obtener más información, consulte .	20 de noviembre de 2018
Nuevas métricas de CloudWatch	Amazon ES ahora ofrece métricas a nivel de nodo y las pestañas Estado del clúster e Estado de instancia en la consola de Amazon ES.	20 de noviembre de 2018
Compatibilidad en China (Pekín	Amazon OpenSearch Service ahora está disponible en la región cn-north-1, donde admite los tipos de instancia s M4, C4 y R4.	17 de octubre de 2018

Cifrado de nodo a nodo	Amazon OpenSearch Service ahora admite el cifrado de nodo a nodo, que mantiene los datos cifrados a medida que Amazon ES los distribuye en todo el clúster.	18 de septiembre de 2018
Actualizaciones de versiones in situ	Amazon OpenSearch Service ahora admite las actualizaciones de las versiones in situ.	14 de agosto de 2018
Compatibilidad con Elasticsearch 6.3 y	Amazon OpenSearch Service ahora es compatible con las versiones 6.3 y 5.6 de Elasticsearch.	14 de agosto de 2018
Registros de errores	Amazon ES ahora permite publicar registros de errores de Elasticsearch a Amazon CloudWatch.	31 de julio de 2018
Instancias reservadas en China (Ningxia	Amazon ES ahora ofrece instancias reservadas en la región China (Ningxia).	29 de mayo de 2018
Instancias reservadas	Ahora Amazon ES ofrece compatibilidad con Instancias reservadas.	7 de mayo de 2018

Actualizaciones anteriores

En la siguiente tabla se describen los cambios importantes en Amazon ES realizados antes de mayo de 2018.

Cambio	Descripción	Fecha
Autenticación de Amazon Cognito para Kibana	Amazon ES ahora ofrece protección de la página de inicio de sesión para Kibana. Para obtener más información, consulte the section called “Descripción de la autenticación de Amazon Cognito para OpenSearch Dashboards”.	2 de abril de 2018
Compatibilidad con Elasticsearch 6.2	Amazon OpenSearch Service ahora es compatible con la versión 6.2 de Elasticsearch.	14 de marzo de 2018
Complemento Korean Analysis	Amazon ES ahora admite una versión optimizada para memoria del complemento del análisis coreano Seunjeon.	13 de marzo de 2018
Actualizaciones de control de acceso instantáneas	Los cambios en las políticas de control de acceso en los dominios de Amazon ES ahora se aplican al instante.	7 de marzo de 2018
Escala de petabytes	Amazon ES admite ahora tipos de instancias I3 y una capacidad total de almacenamiento del dominio de hasta 1,5 PB. Para obtener más información, consulte the section called “Escala de petabytes”.	19 de diciembre de 2017
Cifrado de datos en reposo	Amazon ES ahora admite el cifrado de datos en reposo. Para obtener más información, consulte the section called “Cifrado en reposo”.	7 de diciembre de 2017
Compatibilidad con Elasticsearch 6.0	Amazon ES ahora admite la versión 6.0 de Elasticse arch. Para ver consideraciones e instrucciones sobre las migraciones, consulte the section called “Actualización de dominios”.	6 de diciembre de 2017
Compatibilidad con VPC	Amazon ES ahora permite lanzar dominios dentro de una Amazon Virtual Private Cloud. La compatibilidad con VPC proporciona una capa de seguridad adicional y simplifica las comunicaciones entre Amazon ES y otros servicios dentro de una VPC. Para obtener más informaci ón, consulte the section called “Compatibilidad con VPC”.	17 de octubre de 2017

Cambio	Descripción	Fecha
Publicación de registros lentos	Amazon ES ahora admite la publicación de registros lentos en CloudWatch Logs. Para obtener más información, consulte the section called “Monitoreo de registros”.	16 de octubre de 2017
Compatibilidad con Elasticsearch 5.5	Amazon ES ahora admite la versión 5.5 de Elasticsearch. Ahora puede restablecer las instantáneas automatizadas sin contactar con Soporte y almacenar scripts con la API de `_scripts`.	7 de septiembre de 2017
Compatibilidad con Elasticsearch 5.3	Amazon ES agregó compatibilidad a la versión 5.3 de Elasticsearch.	1 de junio de 2017
Más instancias y capacidad de EBS por clúster	Amazon ES ahora admite hasta 100 nodos y 150 TB de capacidad de EBS por clúster.	5 de abril de 2017
Compatibilidad en Canadá (Central) y la UE (Londres)	Amazon ES agregó compatibilidad en las siguiente s regiones: Canadá (Central), ca-central-1 y Europa (Londres), eu-west-2.	20 de marzo de 2017
Más instancias y mayores volúmenes de EBS	Amazon ES incorpora la compatibilidad con más instancias y mayores volúmenes de EBS.	21 de febrero de 2017
Compatibilidad con Elasticsearch 5.1	Amazon ES agregó compatibilidad a la versión 5.1 de Elasticsearch.	30 de enero de 2017
Compatibilidad con el complemento Phonetic Analysis	Amazon ES lleva incorporada la integración con el complemento Phonetic Analysis, que permite ejecutar consultas con el operador “sounds-like” (sonido similar a) en los datos.	22 de diciembre de 2016
Compatibilidad con la región EE. UU Este (Ohio)	Amazon ES incorpora compatibilidad con la siguiente región: Este de EE. UU. (Ohio), us-east-2.	17 de octubre de 2016

Cambio	Descripción	Fecha
Nueva métrica de rendimiento	Amazon ES agregó una métrica de rendimiento, `ClusterUsedSpace` .	29 de julio de 2016
Compatibilidad con Elasticsearch 2.3	Amazon ES agregó compatibilidad con la versión 2.3 de Elasticsearch.	27 de julio de 2016
Compatibilidad con la región Asia Pacífico (Mumbai)	Amazon ES agregó compatibilidad a la siguiente región: Asia-Pacífico (Mumbai), ap-south-1.	27 de junio de 2016
Más instancias por clúster	Amazon ES aumenta de 10 a 20 el número máximo de instancias (recuento de instancias) por clúster.	18 de mayo de 2016
Compatibilidad con la región Asia Pacífico (Seúl)	Amazon ES agregó compatibilidad a la siguiente región: Asia-Pacífico (Seúl), ap-northeast-2.	28 de enero de 2016
Amazon ES	Versión inicial.	1 de octubre de 2015

AWSGlosario de

Para ver la terminología más reciente de AWS, consulte el Glosario de AWS en la Referencia de Glosario de AWS.