aws

Guía del usuario de

EC2 Image Builder

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

EC2 Image Builder: Guía del usuario de

Copyright © 2026 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 Generador de imágenes? 1
Características de Generador de imágenes 2
Sistemas operativos compatibles 4
Formatos de imágenes admitidas 4
Cuotas predeterminadas 4
AWS Regiones y puntos de enlace 5
Conceptos 5
Precios 8
Relacionado Servicios de AWS 9
Cómo funciona Generador de Imágenes 9
Elementos de las AMI 10
Administración de componentes 11
Recursos creados 12
Distribución 13
Uso compartido de recursos 13
Conformidad 13
Control de versiones semántico 14
Uso de referencias de versión 15
Patrones de referencia de versiones disponibles 16
Recursos que devuelven referencias a versiones 16
Configuración inicial 17
Rol vinculado a servicios de Generador de imágenes 17
Requisitos de configuración 18
Repositorio de contenedor para canalizaciones de imágenes de contenedor 19
Host dedicado para imágenes de macOS 19
Requisitos previos de IAM 19
Requisitos previos del agente de Systems Manager 21
Tutorial: Generador de imágenes 22
Compilación de su primera imagen 22
Creación de un componente personalizado con los parámetros de entrada 23
Utilice los parámetros de Systems Manager con Image Builder 23
Asistente de canalización: crear AMI 23
Paso 1: especificar los detalles de canalización 24
Paso 2: elegir receta 25

Paso 3: definir la configuración de la infraestructura (opcional) 27
Paso 4: definir los ajustes de distribución (opcional) 28
Paso 5: Revisar 28
Paso 6: limpiar 29
Asistente de canalización: crear una imagen de contenedor 31
Paso 1: especificar los detalles de canalización 32
Paso 2: elegir receta 32
Paso 3: definir la configuración de la infraestructura (opcional) 36
Paso 4: definir los ajustes de distribución (opcional) 37
Paso 5: Revisar 37
Paso 6: limpiar 38
Componente personalizado con parámetros 40
Uso de los parámetros en su documento del componente YAML 40
Establecimiento de los parámetros del componente en una receta de Generador de imágenes desde la consola 45
Utilice un parámetro de imagen base en su receta 46
Paso 1: Busque o cree un parámetro del almacén de parámetros 46
Paso 2: Configurar los permisos de IAM (opcional) 47
Paso 3: Cree una receta de imagen que utilice el parámetro 48
Componentes 50
Enumeración y visualización de los componentes 52
Enumeración de componentes de Generador de imágenes 53
Enumere las versiones de compilación de los componentes del AWS CLI 55
Obtenga los detalles de los componentes en AWS CLI 56
Obtenga los detalles de la política de componentes en el AWS CLI 56
Usa AWS Marketplace componentes 57
Descubra los componentes AWS Marketplace 57
Suscríbase a AWS Marketplace los componentes 59
Utilice AWS Marketplace los componentes en una receta 59
Uso de componentes administrados 60
Paquete distribuidor: instalación de aplicaciones de Windows 61
Endurecimiento de CIS 66
Componentes de endurecimiento de STIG 67
Desarrollo de componentes personalizados 113
Creación de un documento del componente YAML 113
Cree componentes personalizados con Generador de imágenes 118

TOE de AWS aplicación de gestión de componentes 129
TOE de AWS descargas 129
Regiones compatibles 131
Referencia de los comandos 133
Preparación manual 139
Uso del marco de documentos de componentes 153
Módulos de acción 207
Configuración de la entrada 325
Recursos de imágenes 330
Enumeración de imágenes y versiones de compilación 330
Enumeración de imágenes 331
Enumeración de imágenes en espera de acción 337
Enumeración de las versiones de compilación de imágenes 339
Ver detalles de recurso de imagen 342
Visualización de detalles de la imagen en la consola de Generador de Imágenes 343
Obtenga los detalles de la política de imágenes en el AWS CLI 351
Creación de imágenes personalizadas con Generador de imágenes 351
Cancela la creación de imágenes desde AWS CLI 354
Importación y exportación de imágenes de máquinas virtuales 354
Importación de una máquina virtual a Generador de imágenes 355
Distribuya los discos de las máquinas virtuales de la compilación de la imagen desde AWS CLI 360
Importe imágenes de disco ISO 360
Sistemas operativos compatibles para la importación de imágenes de disco ISO 361
Requisitos previos 361
Importación de una imagen de disco ISO a Image Builder 362
Lanza una instancia desde la AMI de salida 366
Administración de los resultados de seguridad 367
Configuración de escaneos de seguridad 367
Administración de los resultados de seguridad 369
Limpieza de los recursos de Generador de imágenes 372
Administración del ciclo de vida de imágenes 373
Requisitos previos 374
Creación de un rol de IAM para la administración del ciclo de vida de Generador de imágenes 375

Creación de un rol de IAM para la administración del ciclo de vida de Generador de imágenes entre cuentas ... 376
Enumeración de las políticas de ciclo de vida ... 378
Ver políticas de ciclo de vida ... 381
Visualización de los detalles de la política de ciclo de vida en la consola de Generador de imágenes ... 381
Creación de políticas de ciclo de vida ... 384
Creación de políticas de ciclo de vida (imagen de AMI) ... 384
Creación de políticas de ciclo de vida (imagen de contenedor) ... 388
Funcionamiento de las reglas de ciclo de vida ... 390
Reglas del ciclo de vida de las AMI ... 391
Visualización de los detalles de las reglas de administración del ciclo de vida ... 392
Configuración de imágenes personalizadas ... 393
Recetas ... 393
Enumeración y visualización de recetas de imágenes ... 394
Enumeración y visualización de recetas de contenedor ... 396
Creación de una nueva versión de una receta de imagen ... 398
Creación de una nueva versión de una receta de contenedor ... 413
Eliminar recursos ... 423
Configuraciones de infraestructura ... 423
Enumeración y visualización de las configuraciones de infraestructura ... 425
Crear una configuración de infraestructura ... 426
Actualización de una configuración de infraestructura ... 431
AWS PrivateLink Puntos finales de VPC ... 433
Ajustes de la distribución ... 439
Enumeración y visualización de las configuraciones de la distribución ... 441
Creación y actualización de la distribución de AMI ... 443
Creación y actualización de la distribución de imágenes de contenedores ... 457
Configuración de la distribución entre cuentas de AMI ... 461
Especificación de una plantilla de lanzamiento de AMI ... 470
Utilice una distribución AMI mejorada ... 473
Cómo compartir recursos ... 475
Propietarios del recurso ... 476
Requisitos previos para compartir recursos de Generador de imágenes ... 476
Consumidores de recursos ... 477
Creación de un recurso compartido ... 477

Opción 1: crear un recurso compartido de RAM ... 478
Opción 2: aplicar una política de recursos y convertirla en un recurso compartido existente ... 478
Dejar de compartir un recurso ... 481
Etiquetar recursos ... 482
Etiquete un recurso del AWS CLI ... 483
Quite la etiqueta de un recurso del AWS CLI ... 483
Enumera todas las etiquetas de un recurso específico del AWS CLI ... 484
Eliminación de recursos ... 485
Consola: eliminación de recursos ... 485
Eliminación de recursos (AWS CLI) ... 487
Flujos de trabajo de imágenes ... 489
Marco del flujo de trabajo: etapas ... 490
Acceso a los servicios ... 490
Utilice flujos de trabajo gestionados ... 491
Enumeración de flujos de trabajo de imágenes ... 491
Creación de un flujo de trabajo de imágenes ... 495
Creación de un documento de flujos de trabajo YAML ... 498
Estructura de un documento de flujos de trabajo YAML ... 498
Acciones de paso ... 510
Variables dinámicas ... 554
Instrucciones condicionales ... 560
Gestión de canalizaciones ... 566
Configurar la ejecución de la canalización ... 567
Inhabilita automáticamente una canalización defectuosa ... 567
Configure el registro de canalizaciones ... 568
Ejecute la canalización manualmente ... 569
Uso de expresiones cron ... 570
Enumeración y visualización de las canalizaciones ... 576
Enumere las canalizaciones de imágenes de AWS CLI ... 576
Obtenga los detalles de la canalización de imágenes en la AWS CLI ... 576
Creación y actualización de canalizaciones (AMI) ... 577
Cree una canalización de AMI desde AWS CLI ... 578
Actualización de la canalización desde la consola ... 580
Actualice la canalización desde AWS CLI ... 584
Creación y actualización de canalizaciones (contenedor) ... 586

Cree una canalización a partir del AWS CLI ... 586
Actualización de la canalización desde la consola ... 589
Actualice la canalización desde AWS CLI ... 592
Configuración de flujos de trabajo de canalización ... 594
Definición de grupos de prueba para los flujos de trabajo de prueba ... 595
Establecimiento de los parámetros del flujo de trabajo en una canalización de Generador de imágenes desde la consola ... 596
Especificación del rol de servicio de IAM que Generador de imágenes utiliza para ejecutar las acciones de flujo de trabajo ... 596
Utilice reglas EventBridge ... 597
EventBridge términos ... 597
Consulta EventBridge las reglas de tu pipeline de Image Builder ... 598
Usa EventBridge reglas para programar la construcción de un oleoducto ... 599
Integración de productos y servicios ... 601
Amazon EventBridge ... 604
Mensajes de eventos que envía Generador de imágenes ... 605
Amazon Inspector ... 608
AWS Marketplace ... 609
AWS Marketplace suscripciones en Image Builder ... 610
Descubra los productos de AWS Marketplace imagen desde la consola Image Builder ... 611
Usa un producto AWS Marketplace de imagen en las recetas de Image Builder ... 614
Amazon Simple Notification Service ... 615
Temas de SNS cifrados ... 615
Formato del mensaje de SNS ... 617
Productos de conformidad ... 622
Supervisión de eventos y registros ... 624
CloudTrail registros ... 624
Información sobre Image Builder en CloudTrail ... 624
CloudWatch Registros ... 626
Seguridad en Generador de imágenes ... 628
Protección de datos ... 629
Cifrado y administración de claves ... 630
Almacenamiento de datos ... 636
Privacidad del tráfico entre redes ... 636
Gestión de identidad y acceso ... 637
Público ... 637

Autenticación con identidades 637
Funcionamiento del Generador de imágenes con las políticas y roles de IAM 638
Gestione los perímetros de datos 649
Políticas basadas en identidad 649
Permisos de flujo de trabajo personalizados 653
Políticas basadas en recursos 654
Políticas administradas 656
Roles vinculados a servicios 672
Resolución de problemas 675
Validación de conformidad 677
Resiliencia 678
Seguridad de la infraestructura 678
Administración de parches 679
Prácticas recomendadas 681
Limpieza posterior a la creación necesaria 682
Anule el script de limpieza de Linux. 693
Solucionar problemas de Generador de Imágenes. 697
Solución de problemas de canalizaciones 697
Escenarios de solución de problemas 699
Historial de revisión 705
..... dccxxiii

¿Qué es Generador de imágenes?

EC2 Image Builder es un sistema totalmente gestionado Servicio de AWS que le ayuda a automatizar la creación, la administración y el despliegue de imágenes personalizadas, seguras y de up-to-date servidor. Puede usar Consola de administración de AWS AWS Command Line Interface, o APIs para crear imágenes personalizadas en su Cuenta de AWS.

Usted es el propietario de las imágenes personalizadas que Generador de Imágenes crea en su cuenta. Puede configurar canalizaciones para automatizar las actualizaciones y los parches del sistema para las imágenes de su propiedad. También puede ejecutar un comando independiente para crear una imagen con los recursos de configuración que haya definido.

El asistente de canalización de Generador de Imágenes puede guiarlo por los pasos para crear una imagen personalizada, de la siguiente manera:

Paso 1: especificar los detalles de canalización

- Asigne un nombre a su canalización y añada etiquetas.
- Defina la configuración de los metadatos y el análisis de vulnerabilidades.
- Establezca un cronograma para su canalización.

Paso 2: personalización de la imagen

Puede seleccionar una receta existente o crear una nueva.

- Seleccione una imagen base para sus personalizaciones.
- Añada o elimine software de la imagen base.
- Personalice los ajustes y los scripts con componentes de compilación.
- Seleccione los componentes de prueba que desee ejecutar.

Paso 3: definición del flujo de trabajo

Un flujo de trabajo de imágenes define la secuencia de pasos que Generador de imágenes lleva a cabo durante las etapas de creación y prueba del proceso de creación de imágenes. Forma parte del marco general de flujos de trabajo de Generador de imágenes.

Paso 4: configuración de la infraestructura de compilación

- Seleccione un rol de IAM para asociarlo al perfil de instancia de las instancias que Generador de imágenes lance durante el proceso de creación de la imagen.
- Seleccione uno o varios tipos de instancias que se puedan aplicar en el lanzamiento.
- Seleccione un tema de Amazon Simple Notification Service (SNS) para recibir notificaciones de Generador de imágenes.
- Especifique la VPC, la subred y los grupos de seguridad que se aplican al proceso de creación de imágenes.
- Seleccione la configuración de solución de problemas, como el lugar donde Generador de imágenes escribe los registros y si desea finalizar la instancia de compilación en caso de error (opción predeterminada) o mantenerla en ejecución para seguir solucionando problemas.

Paso 5: definición de la distribución de imágenes

- Seleccione Regiones de AWS dónde distribuye Image Builder su imagen de Amazon Machine (AMI) o imagen de contenedor.
- Si la canalización de Generador de imágenes crea una AMI, Generador de imágenes también admite la siguiente configuración:
  - Seleccione una clave de KMS para el cifrado.
  - Configure el uso compartido de AMI entre Cuentas de AWS and Organizations.
  - Asocie una licencia autoadministrada de License Manager a su imagen distribuida.
  - Configure una plantilla de lanzamiento para su imagen.

Características de Generador de imágenes

EC2 Image Builder ofrece las siguientes funciones:

Aumente la productividad y reduzca las operaciones de creación de up-to-date imágenes y compatibles

Generador de Imágenes reduce la cantidad de trabajo que implica crear y administrar imágenes a escala mediante la automatización de las canalizaciones de compilación. Puede automatizar sus compilaciones proporcionando su preferencia de cronograma de ejecución de compilaciones.

La automatización reduce el costo operativo de mantener el software con los últimos parches del sistema operativo.

Aumento del tiempo de actividad de servicio

Generador de Imágenes proporciona acceso a los componentes de prueba que puede usar para probar las imágenes antes de la implementación. También puede crear componentes de prueba personalizados con Ejecutor y orquestador de tareas de AWS (TOE de AWS) y usarlos. Generador de Imágenes distribuye la imagen solo si todas las pruebas configuradas se han realizado correctamente.

Aumento del nivel de seguridad para las implementaciones

Generador de Imágenes le permite crear imágenes que eliminan la exposición innecesaria a las vulnerabilidades de seguridad de los componentes. Puede aplicar la configuración AWS de seguridad para crear out-of-the-box imágenes seguras que cumplan con los criterios de seguridad internos y del sector. Generador de Imágenes también proporciona conjuntos de ajustes para empresas de sectores regulados. Puede utilizar estos ajustes para compilar imágenes que cumplan con los estándares STIG de forma rápida y sencilla. Para obtener una lista completa de los componentes de STIG disponibles a través de Generador de Imágenes, consulte Componentes de endurecimiento de STIG administrados por Amazon para Generador de imágenes.

Aplicación centralizada y seguimiento del parentesco

Al utilizar integraciones integradas con AWS Organizations, Image Builder le permite aplicar políticas que restringen las cuentas a ejecutar instancias solo cuando están aprobadas AMIs.

Simplificación del uso compartido de recursos entre Cuentas de AWS

EC2 Image Builder se integra con AWS Resource Access Manager (AWS RAM) para permitirte compartir determinados recursos con cualquiera Cuenta de AWS o a través de ellos AWS Organizations. EC2 Los recursos de Image Builder que se pueden compartir son:

- Componentes
- Imágenes
- Recetas de imagen
- Recetas de contenedor

Para obtener más información, consulte Comparta los recursos de Image Builder con AWS RAM.

Sistemas operativos compatibles

Generador de Imágenes es compatible con las siguientes versiones de sistemas operativos:

Sistema operativo/distribución	Versiones compatibles
Amazon Linux	2 y 2023
CentOS	7 y 8
CentOS Stream	8
macOS	12.x (Monterrey), 13.x (Ventura), 14.x (Sonoma), 15.x (Sequoia)
Red Hat Enterprise Linux (RHEL)	7, 8, 9 y 10
SUSE Linux Enterprise Server (SUSE)	12 y 15
Ubuntu	18.04 LTS, 20.04 LTS, 22.04 LTS y 24.04 LTS
Windows Server	2012 R2, 2016, 2019, 2022 y 2025

Formatos de imágenes admitidas

Para sus imágenes personalizadas que crean una Imagen de máquina de Amazon (AMI), puede elegir una AMI existente como punto de partida. En el caso de las imágenes de contenedores de Docker, puede elegir entre imágenes públicas alojadas en DockerHub, imágenes de contenedores existentes en Amazon ECR o imágenes de contenedores gestionados por Amazon como punto de partida.

Cuotas predeterminadas

Para ver las cuotas predeterminadas de Generador de Imágenes, consulte Puntos finales y cuotas de Generador de Imágenes.

AWS Regiones y puntos de enlace

Para ver los puntos de conexión de servicio de Generador de Imágenes, consulte Puntos finales y cuotas de Generador de Imágenes.

Conceptos

Los siguientes términos y conceptos son fundamentales para la comprensión y el uso de EC2 Image Builder.

AMI

Una imagen de máquina de Amazon (AMI) es la unidad básica de despliegue en Amazon EC2 y es uno de los tipos de imágenes que se pueden crear con Image Builder. Una AMI es una imagen de máquina virtual preconfigurada que contiene el sistema operativo (SO) y el software preinstalado para implementar EC2 instancias. Para obtener más información, consulte Imagen de máquina de Amazon (AMI).

Canalización de imágenes

Una canalización de imágenes proporciona un marco de automatización para crear imágenes seguras AMIs y en contenedores. AWS La canalización de imágenes de Generador de Imágenes está asociada a una receta de imágenes o una receta de contenedor que define las fases de compilación, validación y prueba para un ciclo de vida de compilación de imágenes.

Una canalización de imágenes se puede asociar con una configuración de infraestructura que defina dónde se crea la imagen. Puede definir atributos, como el tipo de instancia, las subredes, los grupos de seguridad, el registro y otras configuraciones relacionadas con la infraestructura. También puede asociar la canalización de imágenes con una configuración de distribución para definir cómo desea implementar la imagen.

Administración de imágenes

Una imagen administrada es un recurso de Generador de Imágenes que consta de una AMI o una imagen de contenedor, además de metadatos, como la versión y la plataforma. Las canalizaciones de Generador de Imágenes utilizan la imagen administrada para determinar qué imagen base utilizar para la compilación. En esta guía, las imágenes administradas a veces se denominan “imágenes”; sin embargo, una imagen no es lo mismo que una AMI.

Receta de imagen

Una receta de imagen de Generador de Imágenes es un documento que define la imagen base y los componentes que se aplicarán a la imagen base para producir la configuración deseada de la imagen AMI de salida. Puede usar una receta de imagen para duplicar compilaciones. Las recetas de imágenes de Image Builder se pueden compartir, ramificar y editar mediante el asistente de consola AWS CLI, la o la API. Puede utilizar recetas de imágenes con su software de control de versiones para mantener recetas de imágenes versionadas y compartibles.

Receta de contenedor

Una receta de contenedor de Generador de Imágenes es un documento que define la imagen base y los componentes que se aplican a la imagen base para producir la configuración deseada de la imagen de contenedor de salida. Puede usar una receta de contenedor para duplicar compilaciones. Puede compartir, ramificar y editar las recetas de imágenes de Generador de Imágenes mediante el asistente de consola, la AWS CLI o la API. Puede utilizar recetas de contenedor con su software de control de versiones para mantener las recetas de contenedores versionadas y compartibles.

Imagen base

La imagen base es la imagen seleccionada y el sistema operativo utilizado en el documento de imagen o receta del contenedor, junto con los componentes. La combinación de la imagen base y las definiciones de los componentes producen la configuración deseada para la imagen de salida.

Componentes

Un componente define la secuencia de pasos necesarios para personalizar una instancia antes de la creación de la imagen (un componente de compilación) o para probar una instancia que se lanzó desde la imagen creada (un componente de prueba).

Un componente se crea a partir de un documento YAML o JSON declarativo y de texto simple que describe la configuración del tiempo de ejecución para compilar y validar o probar una instancia producida por su canalización. Los componentes se ejecutan en la instancia mediante una aplicación de administración de componentes. La aplicación de administración de componentes analiza los documentos y ejecuta los pasos deseados.

Una vez creados, uno o más componentes se agrupan mediante una receta de imagen o una receta de contenedor para definir el plan de creación y prueba de una imagen de contenedor o máquina virtual. Puede utilizar componentes públicos que sean propiedad de y estén gestionados por él AWS, o puede crear los suyos propios. Para obtener más información sobre los componentes, consulte Cómo utiliza Image Builder la Ejecutor y orquestador de tareas de AWS aplicación para gestionar los componentes.

Documento componente

Un documento YAML o JSON declarativo y de texto plano que describe la configuración de una personalización que puede aplicar a su imagen. El documento se usa para crear un componente de compilación o prueba.

Etapas de tiempo de ejecución

EC2 Image Builder tiene dos etapas de ejecución: compilación y prueba. Cada etapa de tiempo de ejecución tiene una o más fases con la configuración definida en el documento componente.

Fases de configuración

La siguiente lista muestra las fases que se ejecutan durante las etapas de compilación y prueba:

Etapa de compilación:

Build phase (Fase de compilación)

Una canalización de imágenes comienza con la fase de compilación de la etapa de compilación cuando se ejecuta. Se descarga la imagen base y se aplica la configuración especificada para la fase de compilación del componente para compilar y lanzar una instancia.

Fase de validación

Una vez que Generador de imágenes lanza la instancia y aplica todas las personalizaciones de la fase de compilación, comienza la fase de validación. Durante esta fase, Generador de Imágenes garantiza que todas las personalizaciones funcionen según lo previsto, según la configuración que el componente especifique para la fase de validación. Si la validación de la instancia se realiza correctamente, Generador de Imágenes detiene la instancia, crea una imagen y luego pasa a la etapa de prueba.

Etapa de prueba:

Fase de prueba

Durante esta fase, Generador de Imágenes lanza una instancia a partir de la imagen que creó una vez que la fase de validación finaliza correctamente. Generador de Imágenes ejecuta los componentes de prueba durante esta fase para comprobar que la instancia está en buen estado y funciona según lo esperado.

Fase de prueba del host del contenedor

Una vez que Generador de Imágenes ejecuta la fase de prueba para todos los componentes que ha seleccionado en la receta del contenedor, Generador de Imágenes ejecuta esta fase para los flujos de trabajo del contenedor. La fase de prueba del host del contenedor puede ejecutar pruebas adicionales que validen la administración del contenedor y las configuraciones de tiempo de ejecución personalizadas.

Flujo de trabajo

Los flujos de trabajo definen la secuencia de pasos que Generador de Imágenes realiza al crear una nueva imagen. Todas las imágenes tienen flujos de trabajo de creación, prueba y distribución.

Tipos de flujo de trabajo

`BUILD`

Abarca la configuración de la etapa de compilación para cada imagen creada.

`TEST`

Abarca la configuración de la etapa de prueba para cada imagen creada.

`DISTRIBUTION`

Abarca la configuración de la etapa de distribución para cada imagen creada.

Precios

El uso de EC2 Image Builder para crear AMI personalizadas o imágenes de contenedor no tiene costo alguno. Sin embargo, se aplica un precio estándar para otros servicios que se utilizan en el proceso. En la siguiente lista se incluye el uso de algunos elementos Servicios de AWS que pueden suponer costes al crear, compilar, almacenar y distribuir la AMI personalizada o las imágenes de contenedor, según la configuración.

- Lanzamiento de una instancia EC2
- Almacenamiento de registros en Amazon S3
- Validación de imágenes con Amazon Inspector
- Almacenamiento de instantáneas de Amazon EBS para su AMIs
- Almacenamiento de imágenes de contenedor en Amazon ECR

- Introducción y extracción de imágenes de contenedor dentro y fuera de Amazon ECR
- Si el nivel avanzado de Systems Manager está activado y las EC2 instancias de Amazon se ejecutan con activación local, es posible que se te cobren los recursos a través de Systems Manager

Relacionado Servicios de AWS

EC2 Image Builder utiliza otros Servicios de AWS para crear imágenes, según la configuración de recetas de Image Builder. Para obtener más información sobre la integración de productos y servicios para sus imágenes personalizadas, consulte Integración de productos y servicios en Generador de imágenes.

Cómo funciona EC2 Image Builder

Al utilizar la consola EC2 Image Builder para crear una canalización de imágenes personalizada, el sistema le guiará por los pasos siguientes.

1. Especifique los detalles de la canalización: introduzca la información sobre la canalización, como el nombre, la descripción, las etiquetas y una programación para ejecutar compilaciones automatizadas. Si lo prefiere, puede elegir compilaciones manuales.
2. Elija la receta: elija entre crear una AMI o crear una imagen de contenedor. Para ambos tipos de imágenes de salida, introduzca un nombre y una versión para la receta, seleccione una imagen base y elija los componentes que desee añadir para su creación y prueba. También puede elegir el control de versiones automático para asegurarse de utilizar siempre la última versión disponible del sistema operativo (SO) para su imagen base. Las recetas de contenedores también definen Dockerfiles y el repositorio Amazon ECR de destino para la imagen de contenedor de Docker de salida.

   > **Note**
   >
   > Los componentes son los bloques básicos que consume una receta de imagen o una receta de contenedor. Por ejemplo, paquetes para la instalación, pasos de refuerzo de la seguridad y pruebas. La imagen base y los componentes seleccionados forman una receta de imagen.

3. Defina la configuración de la infraestructura: Image Builder lanza EC2 instancias en su cuenta para personalizar las imágenes y ejecutar pruebas de validación. Los ajustes de configuración de

la infraestructura especifican los detalles de la infraestructura de las instancias que se ejecutarán en su Cuenta de AWS servidor durante el proceso de creación.

4. Defina los ajustes de distribución: elija las regiones de AWS en las que desea distribuir la imagen una vez que la compilación se haya completado y haya superado todas las pruebas. La canalización distribuye automáticamente la imagen a la región en la que se ejecuta la compilación, y puede añadir la distribución de imágenes para otras regiones.

Las imágenes que cree a partir de su imagen base personalizada están en su Cuenta de AWS. Puede configurar la canalización de imágenes para producir versiones actualizadas y parcheadas de la imagen introduciendo un cronograma de creación. Cuando la compilación esté completa, puede recibir una notificación a través de Amazon Simple Notification Service (SNS). Además de producir una imagen final, el asistente de consola Image Builder genera una receta que se puede utilizar con los sistemas de control de versiones existentes y las canalizaciones de integration/continuous despliegue continuo (CI/CD) para una automatización repetible. Puede compartir y crear nuevas versiones de su receta.

Contenidos de la sección

- Elementos de las AMI
- Administración de componentes
- Recursos creados
- Distribución
- Uso compartido de recursos
- Conformidad

Elementos de las AMI

Una imagen de máquina de Amazon (AMI) es una imagen de máquina virtual (VM) preconfigurada que contiene el sistema operativo y el software para implementar EC2 las instancias.

Una AMI incluye los siguientes elementos:

- Una plantilla para el volumen raíz de la máquina virtual. Cuando lanzas una Amazon EC2 VM, el volumen del dispositivo raíz contiene la imagen para arrancar la instancia. Cuando se utiliza el almacén de instancias, el dispositivo raíz es un volumen de almacén de instancias creado a partir de una plantilla de Amazon S3. Para obtener más información, consulta Amazon EC2 Root Device Volume.

- Cuando se utiliza Amazon EBS, el dispositivo raíz es un volumen de EBS creado a partir de una instantánea de EBS.
- Permisos de lanzamiento que determinan qué se puede iniciar VMs con la AMI. Cuentas de AWS
- Datos de Asignación de dispositivos de bloques que especifican los volúmenes que se van a adjuntar a la instancia cuando se lance.
- Un identificador de recursos único para cada región y para cada cuenta.
- Cargas útiles de metadatos, como etiquetas, y propiedades como la región, el sistema operativo, la arquitectura, el tipo de dispositivo raíz, el proveedor, los permisos de inicio, el almacenamiento del dispositivo raíz y el estado de la firma.
- Una firma AMI para las imágenes de Windows para protegerlas contra la manipulación no autorizada. Para obtener más información, consulte Documentos de identidad de instancias.

Administración de componentes

EC2 Image Builder utiliza una aplicación de administración de componentes Ejecutor y orquestador de tareas de AWS (TOE de AWS) que le ayuda a organizar flujos de trabajo complejos, modificar las configuraciones del sistema y probar los sistemas con componentes de script basados en YAML. Como TOE de AWS es una aplicación independiente, no requiere ninguna configuración adicional. Se puede ejecutar en cualquier infraestructura de nube y en las instalaciones. Para empezar a TOE de AWS utilizarla como aplicación independiente, consulte. Configuración manual para desarrollar componentes personalizados con TOE de AWS

Image Builder se utiliza TOE de AWS para realizar todas las actividades en la instancia. Estas incluyen crear y validar la imagen antes de tomar una instantánea y probar la instantánea para asegurarse de que funciona como se espera antes de crear la imagen final. Para obtener más información acerca de cómo Image TOE de AWS Builder administra sus componentes, consulteUso de componentes para personalizar su imagen de Generador de imágenes. Para obtener más información sobre cómo crear componentes con el TOE de AWS, consulte Cómo utiliza Image Builder la Ejecutor y orquestador de tareas de AWS aplicación para gestionar los componentes.

Pruebas de imágenes

Puede usar componentes de TOE de AWS prueba para validar la imagen y asegurarse de que funciona como se espera antes de crear la imagen final.

Por lo general, cada componente de prueba consta de un documento YAML que contiene un script de prueba, un archivo binario de prueba y metadatos de prueba. El script de prueba contiene los

comandos de orquestación para iniciar el archivo binario de prueba, que se puede escribir en cualquier lenguaje compatible con el sistema operativo. Los códigos de estado de salida indican el resultado de la prueba. Los metadatos de la prueba describen la prueba y su comportamiento; por ejemplo, el nombre, la descripción, las rutas al archivo binario de la prueba y la duración prevista.

Recursos creados

Al crear una canalización, no se crea ningún recurso externo a Generador de Imágenes, a menos que se cumpla lo siguiente:

- Cuando se crea una imagen mediante el cronograma de canalización.
- Cuando se elige Ejecutar canalización en el menú Acciones de la consola de Generador de Imágenes.
- Cuando ejecutas cualquiera de estos comandos desde la API o AWS CLI: StartImagePipelineExecution o CreateImage

Los siguientes recursos se crean durante el proceso de creación de la imagen:

Canalizaciones de imágenes AMI

- EC2 instancia (temporal)
- Asociación de inventario de Systems Manager (a través de Systems Manager State Manager si `EnhancedImageMetadata` está habilitada) en la EC2 instancia
- Amazon EC2 AMI
- La instantánea de Amazon EBS asociada a la AMI de Amazon EC2

Canalizaciones de imágenes de contenedor

- El contenedor Docker se ejecuta en una EC2 instancia (temporal)
- La asociación de inventario de Systems Manager (a través de Systems Manager (State Manager) `EnhancedImageMetadata` está habilitada) en la EC2 instancia
- Imagen de contenedor de Docker
- Dockerfile

Una vez creada la imagen, se eliminan todos los recursos temporales.

Distribución

EC2 Image Builder puede distribuir AMIs o almacenar imágenes en cualquier AWS región. La imagen se copia en cada región que especifique en la cuenta utilizada para crear la imagen.

Para las imágenes de salida de la AMI, puede definir los permisos de lanzamiento de la AMI para controlar cuáles Cuentas de AWS están autorizados a lanzar EC2 instancias con la AMI creada. Por ejemplo, puede hacer que la imagen sea privada, pública o compartirla con cuentas específicas. Si distribuye la AMI a otras regiones y define permisos de lanzamiento para otras cuentas, los permisos de lanzamiento se propagan a todas las regiones en las que se distribuye la AMI. AMIs

También puede usar su AWS Organizations cuenta para imponer limitaciones a las cuentas de los miembros a la hora de lanzar instancias únicamente si están aprobadas y cumplen AMIs con las normas. Para obtener más información, consulta Cómo administrar Cuentas de AWS en tu organización.

Para actualizar los ajustes de distribución mediante la consola Generador de Imágenes, siga los pasos para Creación de una nueva versión de una receta de imagen de la consola o Creación de una nueva versión de una receta de contenedor con la consola.

Uso compartido de recursos

Para compartir componentes, recetas o imágenes con otras cuentas o dentro de ellas AWS Organizations, consulteComparta los recursos de Image Builder con AWS RAM.

Conformidad

Para los puntos de referencia del Center for Internet Security (CIS), EC2 Image Builder utiliza Amazon Inspector para evaluar la exposición, las vulnerabilidades y las desviaciones de las mejores prácticas y los estándares de conformidad. Por ejemplo, Image Builder evalúa la accesibilidad no deseada de la red, la conectividad pública a Internet sin parches CVEs y la activación del inicio de sesión raíz remoto. Amazon Inspector se ofrece como un componente de prueba que puede añadir a su receta de imagen. Para obtener más información sobre Amazon Inspector, consulta la Guía del usuario de Amazon Inspector. Para obtener más información, consulte los puntos de referencia del Center for Internet Security (CIS).

Generador de Imágenes proporciona componentes de endurecimiento de STIG para crear de manera más eficiente imágenes compatibles con los estándares de referencia de STIG. Estos componentes de STIG escanean en busca de configuraciones incorrectas y ejecutan un script de

corrección. El uso de componentes que cumplen con STIG no conlleva cargos adicionales. Para obtener una lista completa de los componentes de STIG disponibles a través de Generador de Imágenes, consulte Componentes de endurecimiento de STIG administrados por Amazon para Generador de imágenes.

Control de versiones semántico en Generador de imágenes

Image Builder utiliza el control de versiones semántico para organizar los recursos y garantizar que sean únicos. IDs La versión semántica tiene cuatro nodos:

<major>. *<minor>*. *<patch>*/<build>

Puede asignar valores a los tres primeros y puede filtrar en todos ellos.

El control de versiones semántico se incluye en el nombre de recurso de Amazon (ARN) de cada objeto, en el nivel que se aplica a ese objeto de la siguiente manera:

1. Versionless ARNs y Name ARNs no incluyen valores específicos en ninguno de los nodos. Los nodos se omiten por completo o se especifican como caracteres comodín, por ejemplo, x.x.x.
2. <major>La versión ARNs tiene solo los tres primeros nodos:. <minor>. <patch>
3. La versión de compilación ARNs tiene los cuatro nodos y apunta a una compilación específica para una versión específica de un objeto.

Asignación: a los tres primeros nodos puedes asignar cualquier valor entero positivo, o cero, con un límite superior de 2^30-1, o 1073741823 para cada nodo. El constructor de imágenes asigna automáticamente el número de construcción al cuarto nodo.

Patrones: puede usar cualquier patrón numérico que cumpla con los requisitos de asignación de los nodos que puede asignar. Por ejemplo, puede elegir un patrón de versión de software, como 1.0.0 o una fecha como 2021.01.01.

Selección: con el control de versiones semántico, tiene la flexibilidad de usar comodines (x) para especificar las versiones o los nodos más recientes al seleccionar la imagen base o los componentes de su receta. Cuando se usa un comodín en cualquier nodo, todos los nodos a la derecha del primer comodín también deben ser comodines.

Por ejemplo, dadas las siguientes versiones recientes: 2.2.4, 1.7.8 y 1.6.8, la selección de versiones mediante caracteres comodín produce los siguientes resultados:

- `x.x.x` = 2.2.4
- `1.x.x` = 1.7.8
- `1.6.x` = 1.6.8
- `x.2.x` no es válido y produce un error.
- `1.x.8` no es válido y produce un error.

Uso de referencias de versión

Las referencias de versión son cadenas de ready-to-use ARN que incorporan patrones comodín basados en la versión semántica del recurso que ha creado o recuperado. En lugar de escribir código personalizado para analizar ARNs e insertar caracteres comodín, Image Builder hace este trabajo por usted.

Al crear o recuperar recursos de Image Builder, Image Builder proporciona automáticamente patrones de versiones preconstruidas ARNs con caracteres comodín en el `latestVersionReferences` objeto. Esto elimina la necesidad de analizar y reconstruir manualmente los recursos ARNs cuando se quiere hacer referencia a los recursos mediante patrones de control de versiones comodín.

Por ejemplo, al crear un componente con version`1.2.3`, Image Builder devuelve:

```
{
    "componentBuildVersionArn": "arn:aws:imagebuilder:us-west-2:123456789012:component/
my-component/1.2.3/1",
    "latestVersionReferences": {
        "latestVersionArn": "arn:aws:imagebuilder:us-west-2:123456789012:component/my-
component/x.x.x",
        "latestMajorVersionArn": "arn:aws:imagebuilder:us-
west-2:123456789012:component/my-component/1.x.x",
        "latestMinorVersionArn": "arn:aws:imagebuilder:us-
west-2:123456789012:component/my-component/1.2.x",
        "latestPatchVersionArn": "arn:aws:imagebuilder:us-
west-2:123456789012:component/my-component/1.2.3"
    }
}
```

Patrones de referencia de versiones disponibles

El `latestVersionReferences` objeto contiene cuatro patrones de ARN:

- latestVersionArn (x.x.x): siempre se resuelve con la última versión absoluta.
- atestMajorVersionArn (1.x.x): utiliza las últimas versiones secundarias y de parche dentro de una versión principal.
- latestMinorVersionArn (1.2.x): se resuelve con la última versión del parche dentro de una versión secundaria específica.
- latestPatchVersionArn (1.2.3): hace referencia a una versión semántica específica y elige la versión de compilación más reciente para los recursos que admiten varias versiones de compilación.

Recursos que devuelven referencias a versiones

Las referencias de versión se devuelven por `Create` y `Get` APIs para todos los recursos de Image Builder versionados:

- Componentes:, `CreateComponent` `GetComponent`
- Recetas de imágenes -`CreateImageRecipe`, `GetImageRecipe`
- Recetas en contenedores -`CreateContainerRecipe`, `GetContainerRecipe`
- Imágenes -`CreateImage`, `GetImage`
- Flujos de trabajo -`CreateWorkflow`, `GetWorkflow`

Nota: Para los flujos de trabajo gestionados por Image Builder, solo `latestVersionArn (x.x.x)` se devuelve, ya que Image Builder requiere que utilice siempre la última versión de los flujos de trabajo gestionados por Image Builder.

Preparación para crear imágenes personalizadas con Generador de imágenes

Antes de crear imágenes con EC2 Image Builder, compruebe que cumple los siguientes requisitos previos para crear una canalización de imágenes. A menos que se especifique lo contrario, estos requisitos previos se requieren para todos los tipos de canalizaciones.

Requisitos previos

- Rol vinculado a servicios de Generador de imágenes
- Requisitos de configuración
- Repositorio de contenedor para canalizaciones de imágenes de contenedor
- Host dedicado para imágenes de macOS
- Requisitos previos de IAM
- Requisitos previos del agente de Systems Manager

Una vez que haya cumplido los requisitos previos, podrá administrar EC2 Image Builder desde cualquiera de las siguientes interfaces.

- EC2 Consola Image Builder
- Comandos de Image Builder en AWS CLI
- EC2 Referencia de la API Image Builder
- AWS SDKs y herramientas

Rol vinculado a servicios de Generador de imágenes

EC2 Image Builder utiliza un rol vinculado a un servicio para conceder permisos a otros AWS servicios en su nombre. No necesita crear manualmente un rol vinculado a servicios. Cuando crea su primer recurso de Image Builder en la consola AWS de administración AWS CLI, la o la AWS API, Image Builder crea el rol vinculado al servicio por usted. Para obtener más información sobre el rol vinculado a servicios que Generador de Imágenes crea en su cuenta, consulte Uso de roles vinculados a servicios de IAM para Generador de imágenes.

Requisitos de configuración

- Generador de Imágenes es compatible con AWS PrivateLink. Para obtener más información sobre cómo configurar el punto de conexión de VPC para Generador de Imágenes, consulte Image Builder y puntos finales AWS PrivateLink de VPC de interfaz.
- Las instancias que Image Builder utiliza para crear imágenes de contenedores deben tener acceso a Internet para descargarlas AWS CLI de Amazon S3 y para descargar una imagen base del repositorio de Docker Hub, si corresponde. Image Builder usa el AWS CLI para obtener el Dockerfile de la receta del contenedor, donde se almacena como datos.
- Las instancias que Generador de Imágenes utiliza para crear imágenes y ejecutar pruebas deben tener acceso al servicio Systems Manager. Los requisitos de instalación dependen del sistema operativo.

  Para ver los requisitos de instalación de la imagen base, seleccione la pestaña que corresponda al sistema operativo de la imagen base.

  Linux

  Para las instancias de Amazon EC2 Linux, Image Builder instala el agente de Systems Manager en la instancia de compilación si aún no está presente y lo elimina antes de crear la imagen.

  Windows

  Image Builder no instala el agente de Systems Manager en las instancias de compilación de Amazon EC2 Windows Server. Si la imagen base no venía preinstalada con Systems Manager Agent, debe lanzar una instancia desde la imagen de origen, instalar Systems Manager manualmente en la instancia y crear una nueva imagen base a partir de la instancia.

  Para instalar manualmente el agente de Systems Manager en su instancia de Amazon EC2 Windows Server, consulte Instalación manual del agente de Systems Manager en EC2 instancias de Windows Server en la Guía del AWS Systems Manager usuario.

Repositorio de contenedor para canalizaciones de imágenes de contenedor

Para las canalizaciones de imágenes de contenedor, la receta define la configuración de las imágenes de Docker que se producen y almacenan en el repositorio de contenedor de destino. Debe crear el repositorio de destino antes de crear la receta de contenedor para su imagen de Docker.

Generador de Imágenes utiliza Amazon ECR como repositorio de destino para las imágenes de contenedor. Para crear un repositorio de Amazon ECR, siga los pasos descritos en Crear un repositorio de la Guía del usuario del registro de contendores de Amazon Elastic.

Host dedicado para imágenes de macOS

Las instancias de Amazon EC2 Mac requieren un host dedicado que se ejecute en un tipo de instancia metálica. Antes de crear una imagen de macOS personalizada, debe asignar un host dedicado a su cuenta. Para obtener más información sobre las instancias de Mac y una lista de los tipos de instancias compatibles de forma nativa con el sistema operativo macOS, consulta las instancias de Amazon EC2 Mac en la Guía del EC2 usuario de Amazon.

Cuando haya creado un host dedicado, podrá configurar los ajustes en el recurso de configuración de infraestructura de su imagen. La configuración de la infraestructura incluye propiedades de ubicación, en las que puede especificar el host, el grupo de ubicación de hosts o la zona de disponibilidad a los que deben ir las instancias que se lanzan desde la imagen.

Requisitos previos de IAM

El rol de IAM que asocie a su perfil de instancia debe tener permisos para ejecutar los componentes de compilación y prueba incluidos en su imagen. Las siguientes políticas del rol de IAM se deben asociar al rol de IAM que está asociado al perfil de instancia:

- EC2InstanceProfileForImageBuilder
- EC2InstanceProfileForImageBuilderECRContainerBuilds
- Amazon SSMManaged InstanceCore

Si configura el registro, el perfil de instancia especificado en la configuración de la infraestructura debe tener permisos `s3:PutObject` para el bucket de destino (`arn:aws:s3:::BucketName/*`). Por ejemplo:

JSON

```
{
 "Version":"2012-10-17",
 "Statement": [
  {
   "Sid": "UploadFileToS3Bucket",
   "Effect": "Allow",
   "Action": "s3:PutObject",
   "Resource": "arn:aws:s3:::bucket-name/*"
  }
 ]
}
```

Asociar política

Los siguientes pasos lo guían en el proceso de asociar las políticas de IAM a un rol de IAM para conceder los permisos anteriores.

1. Inicie sesión en la consola AWS de administración y abra la consola de IAM en https://console.aws.amazon.com/iam/.
2. En el panel de navegación izquierdo, elija Políticas.
3. Filtre la lista de políticas con EC2InstanceProfileForImageBuilder
4. Seleccione la viñeta junto a la política y, en la lista desplegable Acciones de política, seleccione Asociar.
5. Seleccione el nombre del rol de IAM al que se asociará la política.
6. Elija Asociar política.
7. Repite los pasos 3 a 6 para las SSMManaged InstanceCore políticas EC2InstanceProfileForImageBuilderECRContainerBuildsy de Amazon.

> **Note**
>
> Si desea copiar una imagen creada con Generador de Imágenes a otra cuenta, debe crear el rol de `EC2ImageBuilderDistributionCrossAccountRole` en todas las cuentas de destino y asociar la política administrada Política

Ec2ImageBuilderCrossAccountDistributionAccess al rol. Para obtener más información, consulte Comparta los recursos de Image Builder con AWS RAM.

Requisitos previos del agente de Systems Manager

EC2 Image Builder ejecuta el agente AWS Systems Manager(Systems Manager) en las EC2 instancias que lanza para crear y probar la imagen. Generador de Imágenes recopila información adicional sobre la instancia utilizada durante la fase de compilación con Systems Manager Inventory. Esta información incluye el nombre y la versión del sistema operativo (OS), así como la lista de paquetes y sus versiones respectivas, según lo indicado por su sistema operativo.

Para dejar de recopilar esta información, seleccione el método que mejor se adapte a su entorno preferido:

- Consola de Generador de imágenes: desactive la casilla Habilitar la recopilación de metadatos mejorada.
- AWS CLI: especifique la opción `--no-enhanced-image-metadata-enabled`.
- Image Builder API o SDKs: defina el `enhancedImageMetadataEnabled` parámetro en`false`.

Generador de Imágenes utiliza `RunCommand` para enviar acciones a la instancia de compilación y prueba como parte del flujo de trabajo de compilación y prueba de imágenes. No puede optar por dejar de usar `RunCommand` para enviar acciones a su instancia de compilación y prueba.

Creación de imágenes personalizadas con los tutoriales de Generador de imágenes

Hay muchas maneras de crear imágenes y componentes personalizados con EC2 Image Builder. Los tutoriales le ayudan a aprender los conceptos clave de Generador de imágenes. Cada tutorial presenta un caso de uso con pasos que puede seguir por primera vez. Las instrucciones utilizan los valores predeterminados siempre que es posible para ayudarlo a aprender el proceso general. Después de usar uno de los tutoriales, puede explorar más formas de personalizar sus propias imágenes.

Compilación de su primera imagen

Los siguientes tutoriales le muestran cómo crear su primera imagen con el asistente de consola de Generador de imágenes. Al final del tutorial, habrá creado el siguiente conjunto de recursos de Generador de imágenes. El último paso del tutorial es eliminar los recursos que ha creado.

- Una receta de imagen para su Imagen de máquina de Amazon (AMI) o una receta de contenedor para su imagen de contenedor.
- Un recurso de configuración de infraestructura con los ajustes predeterminados.
- Un recurso de configuración de distribución con la configuración predeterminada que distribuye la salida a la región de origen (su cuenta en la región que utiliza para ejecutar el asistente de consola).
- Una canalización de imágenes que utiliza los recursos enumerados para crear la imagen de salida con los flujos de trabajo de creación de imágenes predeterminados.
- Una AMI de salida o una imagen de contenedor.

Tutoriales del asistente de consola

- Asistente de canalización: crear AMI
- Asistente de canalización: crear una imagen de contenedor

Creación de un componente personalizado con los parámetros de entrada

El siguiente tutorial muestra cómo crear un componente personalizado que defina los parámetros de entrada y, a continuación, establecer los valores de la receta de Generador de imágenes.

Componente personalizado con parámetros

Utilice los parámetros de Systems Manager con Image Builder

El siguiente tutorial muestra cómo crear un AWS Systems Manager parámetro del almacén de parámetros y utilizarlo en una receta de imagen.

Utilice un parámetro de imagen base en su receta

También puede usar los parámetros del almacén de parámetros en la configuración de distribución de AMI para almacenar el ID de la imagen de salida y en los componentes personalizados. Para obtener más información, consulte Creación y actualización de ajustes de distribución de AMI las distribuciones y Utilice los parámetros del almacén de parámetros de Systems Manager los componentes personalizados.

Tutorial: creación de una canalización de imágenes con una AMI de salida desde el asistente de la consola de Generador de Imágenes

En este tutorial, se explica cómo crear una canalización automatizada para crear y mantener una EC2 imagen personalizada de Image Builder mediante el asistente de consola Crear canalización de imágenes. Para ayudarle a realizar los pasos de forma eficaz, se utilizan los ajustes predeterminados cuando están disponibles y se omiten las secciones opcionales.

Creación de un flujo de trabajo de canalizaciones de imágenes

- Paso 1: especificar los detalles de canalización
- Paso 2: elegir receta
- Paso 3: definir la configuración de la infraestructura (opcional)
- Paso 4: definir los ajustes de distribución (opcional)

- Paso 5: Revisar
- Paso 6: limpiar

Paso 1: especificar los detalles de canalización

1. Abra la consola EC2 de Image Builder en https://console.aws.amazon.com/imagebuilder/.
2. Para empezar a crear la canalización, seleccione Crear canalizaciones de imágenes.
3. En la sección General, introduzca el nombre de su canalización (obligatorio).

   > **Tip**
   >
   > La recopilación mejorada de metadatos está activada de forma predeterminada. Para garantizar la compatibilidad entre los componentes y las imágenes base, manténgala activada.

4. En la sección Crear programación, puede mantener los valores predeterminados de las opciones de programación. Tenga en cuenta que la zona horaria que se muestra en el horario predeterminado es la hora universal coordinada (UTC). Para obtener más información sobre la hora UTC y encontrar el desfase de su zona horaria, consulte Abreviaturas de zonas horarias: lista mundial.

   Para configurar las actualizaciones de dependencias, seleccione la opción Ejecutar la canalización a la hora programada si hay actualizaciones de dependencias. Esta configuración hace que la canalización compruebe si hay actualizaciones antes de iniciar la compilación. Si no hay actualizaciones, se omite la compilación programada de la canalización.

   > **Note**
   >
   > Para garantizar que su proceso reconozca las actualizaciones y compilaciones de dependencias según lo previsto, debe usar el control de versiones semántico (x.x.x) para la imagen base y los componentes. Para obtener más información sobre el control de versiones semántico para los recursos de Generador de Imágenes, consulte Control de versiones semántico en Generador de imágenes.

5. Para continuar en el paso siguiente, seleccione Siguiente.

Paso 2: elegir receta

1. El valor predeterminado de Generador de Imágenes es Usar la receta existente en la sección Receta. Por primera vez, elija la opción Crear nueva receta.
2. En la sección Tipo de imagen, elija la opción Imagen de máquina de Amazon (AMI) para crear una canalización de imágenes que produzca y distribuya una AMI.
3. En la sección General, introduzca las siguientes casillas obligatorias:

   - Nombre: el nombre de su receta
   - versión: la versión de su receta (use el formato) <major>.<minor>.<patch>, donde major, minor y patch son valores enteros). Por lo general, las recetas nuevas comienzan con `1.0.0`.
4. En la sección Imagen de origen, mantenga los valores predeterminados para Seleccionar imagen, Sistema operativo (SO) de la imagen y Origen de la imagen. Esto da como resultado una lista de Linux AMIs gestionados por Amazon. Para este tutorial, selecciona la `Amazon Linux 2 x86` imagen.

   a. En el menú desplegable Nombre de la imagen, elija una imagen.

   b. Mantenga el valor predeterminado para las opciones de control de versiones automático (utilice la última versión del sistema operativo disponible).

   > **Note**
   >
   > Esta configuración garantiza que la canalización utilice el control de versiones semántico para la imagen base, a fin de detectar las actualizaciones de dependencias en los trabajos programados automáticamente. Para obtener más información sobre el control de versiones semántico para los recursos de Generador de Imágenes, consulte Control de versiones semántico en Generador de imágenes.
5. En la sección Configuración de instancias, conserve los valores predeterminados de Systems Manager Agent. Esto hace que Generador de Imágenes conserve el agente de Systems Manager una vez finalizadas la compilación y las pruebas, para incluir Systems Manager Agent en la nueva imagen.

   Mantenga los datos de usuario en blanco para este tutorial. Puede utilizar esta área en otras ocasiones para proporcionar comandos o un script de comandos para que se ejecuten al lanzar la instancia de compilación. Sin embargo, reemplaza cualquier comando que Generador de Imágenes haya agregado para garantizar que Systems Manager esté instalado. Cuando lo

utilice, asegúrese de que el agente de Systems Manager esté preinstalado en la imagen base o de que incluya la instalación en los datos de usuario.

6. En la sección Componentes, debe elegir al menos un componente de compilación.

   En el panel Construir componentes, elija Agregar componentes de construcción y seleccione un elemento `Amazon managed` de la lista de filtros de propietarios de componentes. Se abre un panel de selección en la parte derecha de la interfaz de la consola, donde puede buscar y filtrar los componentes que están disponibles.

   Para este tutorial, elija un componente que actualice Linux con las actualizaciones de seguridad más recientes, de la siguiente manera:

   a. Filtre los resultados introduciendo la palabra `update` en la barra de búsqueda que se encuentra en la parte superior del panel.

   b. Seleccione la casilla de verificación del componente de compilación `update-linux`.

   c. Mantenga las opciones de control de versiones predeterminadas (utilice la última versión disponible).

      > **Note**
      >
      > Esta configuración garantiza que la canalización utilice el control de versiones semántico para el componente seleccionado a fin de detectar las actualizaciones de dependencias en los trabajos programados automáticamente. Para obtener más información sobre el control de versiones semántico para los recursos de Generador de Imágenes, consulte Control de versiones semántico en Generador de imágenes.

   d. Seleccione Añadir a la receta para añadir el componente a la receta. Esto cierra el panel de selección de componentes.

   e. De vuelta en el panel Construir componentes, se muestra el componente que ha añadido.

7. Reordenar los componentes (opcional)

   Si ha elegido más de un componente para incluirlos en la imagen, puede utilizar la drag-and-drop acción para reorganizarlos en el orden en el que deberían ejecutarse durante el proceso de creación.

> **Note**
>
> Los componentes de endurecimiento del CIS no siguen las reglas de ordenación de componentes estándar de las recetas de Generador de Imágenes. Los componentes de endurecimiento del CIS siempre se ejecutan en último lugar para garantizar que las pruebas de referencia se ejecuten con la imagen de salida.

a. Repite los pasos anteriores para añadir el `update-linux-kernel-5` componente a la receta.
b. El componente que acaba de añadir tiene un parámetro de entrada para la versión del núcleo. Para ampliar la configuración de las opciones de control de versiones o los parámetros de entrada, puede seleccionar la flecha junto al nombre de la configuración. Para ampliar todos los ajustes de todos los componentes seleccionados, puede activar y desactivar el interruptor Expandir todo. Para obtener más información sobre el uso de parámetros de entrada en los componentes y su configuración en las recetas, consulte Tutorial: creación de un documento del componente con los parámetros de entrada.
c. Elija uno de los componentes y arrástrelo hacia arriba o hacia abajo para cambiar el orden en el que se ejecutarán los componentes.
d. Para eliminar el componente `update-linux-kernel-5`, seleccione X en la esquina superior derecha del cuadro de componentes.

   Repita este paso para eliminar cualquier otro componente que haya agregado, dejando solo el `update-linux` componente seleccionado.

8. Para continuar en el paso siguiente, seleccione Siguiente.

Paso 3: definir la configuración de la infraestructura (opcional)

Image Builder lanza EC2 instancias en su cuenta para personalizar las imágenes y ejecutar pruebas de validación. Los ajustes de configuración de la infraestructura especifican los detalles de la infraestructura de las instancias que se ejecutarán en su Cuenta de AWS servidor durante el proceso de creación.

En la sección Configuración de infraestructura, las opciones de configuración están predeterminadas en `Create infrastructure configuration using service defaults`. Esto crea un rol de IAM y un perfil de instancia asociado para las instancias de EC2 compilación y prueba que se

utilizan para configurar la imagen. Para obtener más información sobre los ajustes de configuración de la infraestructura, consulte CreateInfrastructureConfigurationla Referencia de la API de EC2 Image Builder.

En este tutorial, utilizamos la configuración predeterminada.

> **Note**
>
> Para especificar una subred para usarla en una VPC privada, puede crear su propia configuración de infraestructura personalizada o usar los ajustes que ya haya creado.

- Para continuar en el paso siguiente, seleccione Siguiente.

Paso 4: definir los ajustes de distribución (opcional)

Las configuraciones de distribución incluyen el nombre de la AMI de salida, la configuración regional específica para el cifrado, los permisos de lanzamiento y Cuentas de AWS las organizaciones y unidades organizativas (OUs) que pueden lanzar la AMI de salida y las configuraciones de licencia.

En la sección Ajustes de distribución, las opciones de configuración están predeterminadas en `Create distribution settings using service defaults`. Esta opción distribuirá la AMI de salida a la región actual. Para obtener más información acerca de los ajustes de su distribución, consulte Administrar los ajustes de la distribución de Generador de imágenes.

En este tutorial, utilizamos la configuración predeterminada.

- Para continuar en el paso siguiente, seleccione Siguiente.

Paso 5: Revisar

La sección de revisión muestra todos los ajustes que ha configurado. Para editar la información de una sección determinada, presione el botón Editar situado en la esquina superior derecha de la sección de pasos. Por ejemplo, si quiere cambiar el nombre de la canalización, presione el botón Editar en la esquina superior derecha de la sección Paso 1: detalles de la canalización.

1. Cuando haya revisado la configuración, seleccione Crear canalización para crear su canalización.

2. Puede ver los mensajes de éxito o falla en la parte superior de la página, a medida que se crean los recursos para los ajustes de la distribución, la configuración de la infraestructura, la nueva receta y la canalización. Para ver los detalles de un recurso, incluido el identificador del recurso, seleccione Ver detalles.
3. Después de ver los detalles de un recurso, puede ver los detalles de otros recursos seleccionando el tipo de recurso en el panel de navegación. Por ejemplo, para ver los detalles de su nueva canalización, seleccione Canalizaciones de imágenes en el panel de navegación. Si la compilación se realizó correctamente, la nueva canalización se mostrará en la lista de canalizaciones de imágenes.

Paso 6: limpiar

Su entorno Generador de Imágenes, al igual que su hogar, necesita un mantenimiento habitual para ayudarlo a encontrar lo que necesita y completar sus tareas sin tener que preocuparse por el desorden. Asegúrese de limpiar periódicamente los recursos temporales que creó para las pruebas. De lo contrario, es posible que se olvide de esos recursos y, más adelante, no recuerde para qué se utilizaron. Para entonces, es posible que no esté claro si puede deshacerse de ellos de manera segura.

> **Tip**
>
> Para evitar errores de dependencia al eliminar recursos, asegúrese de eliminarlos en el siguiente orden:
>
> 1. Canalización de imágenes
> 2. Receta de imagen
> 3. Todos los recursos restantes

Siga los pasos de esta sección para eliminar los recursos que ha creado en este tutorial:

Eliminación de la canalización

1. Para ver una lista de las canalizaciones de compilación creadas en su cuenta, seleccione las canalizaciones de imágenes en el panel de navegación.
2. Seleccione la casilla de verificación junto al nombre de canalización para seleccionar la canalización que desea eliminar.

3. En la parte superior del panel Canalizaciones de imágenes, en el menú Acciones, seleccione Eliminar.
4. Para confirmar la eliminación, ingrese `Delete` en el cuadro y seleccione Eliminar.

Eliminación de la receta

1. Para ver una lista de las recetas creadas en su cuenta, seleccione Recetas de imagen en el panel de navegación.
2. Seleccione la casilla de verificación junto al nombre de la receta para seleccionar la receta que desea eliminar.
3. En la parte superior del panel Recetas de imagen, en el menú Acciones, seleccione Eliminar receta.
4. Para confirmar la eliminación, ingrese `Delete` en el cuadro y seleccione Eliminar.

Eliminación de configuración de infraestructura

1. Para ver una lista de las configuraciones de infraestructura creadas en su cuenta, seleccione Configuración de infraestructura en el panel de navegación.
2. Seleccione la casilla de verificación junto al nombre de la configuración para seleccionar la configuración de infraestructura que desea eliminar.
3. En la parte superior del panel Configuraciones de infraestructura, seleccione Eliminar.
4. Para confirmar la eliminación, ingrese `Delete` en el cuadro y seleccione Eliminar.

Eliminación de ajustes de distribución

1. Para ver una lista de los ajustes de distribución creados en su cuenta, seleccione Ajustes de distribución en el panel de navegación.
2. Seleccione la casilla de verificación junto al nombre de la configuración para seleccionar los ajustes de distribución que creó para este tutorial.
3. En la parte superior del panel Ajustes de distribución, seleccione Eliminar.
4. Para confirmar la eliminación, ingrese `Delete` en el cuadro y seleccione Eliminar.

Eliminación de la imagen

Siga estos pasos para comprobar que ha eliminado cualquier imagen que se haya creado de la canalización del tutorial. No es probable que este tutorial cree una imagen a menos que haya transcurrido suficiente tiempo desde que creó la canalización para que se ejecute, de acuerdo con el cronograma de compilación.

1. Para ver una lista de las imágenes creadas en su cuenta, seleccione Imágenes en el panel de navegación.
2. Elija la versión de la imagen que desea eliminar. Esto abre la página Versiones de compilación de imágenes.
3. Seleccione la casilla de verificación junto a la versión de cualquier imagen que desea eliminar. Puede seleccionar más de una versión de imagen a la vez.
4. En la parte superior del panel Versiones de compilación de imágenes, seleccione Eliminar versión.
5. Para confirmar la eliminación, ingrese `Delete` en el cuadro y seleccione Eliminar.

Tutorial: creación de una canalización de imágenes con una imagen de contenedor de Docker de salida desde el asistente de consola de Generador de imágenes

En este tutorial, se explica cómo crear una canalización automatizada para crear y mantener una imagen de Docker personalizada de EC2 Image Builder mediante el asistente de consola Crear canalización de imágenes. Para ayudarle a realizar los pasos de forma eficaz, se utilizan los ajustes predeterminados cuando están disponibles y se omiten las secciones opcionales.

Creación de un flujo de trabajo de canalizaciones de imágenes

- Paso 1: especificar los detalles de canalización
- Paso 2: elegir receta
- Paso 3: definir la configuración de la infraestructura (opcional)
- Paso 4: definir los ajustes de distribución (opcional)
- Paso 5: Revisar
- Paso 6: limpiar

Paso 1: especificar los detalles de canalización

1. Abra la consola EC2 de Image Builder en https://console.aws.amazon.com/imagebuilder/.
2. Para empezar a crear la canalización, seleccione Crear canalizaciones de imágenes.
3. En la sección General, introduzca el nombre de su canalización (obligatorio).
4. En la sección Crear programación, puede mantener los valores predeterminados de las opciones de programación. Tenga en cuenta que la zona horaria que se muestra en el horario predeterminado es la hora universal coordinada (UTC). Para obtener más información sobre la hora UTC y encontrar el desfase de su zona horaria, consulte Abreviaturas de zonas horarias: lista mundial.

   Para configurar las actualizaciones de dependencias, seleccione la opción Ejecutar la canalización a la hora programada si hay actualizaciones de dependencias. Esta configuración hace que la canalización compruebe si hay actualizaciones antes de iniciar la compilación. Si no hay actualizaciones, se omite la compilación programada de la canalización.

   > **Note**
   >
   > Para garantizar que su proceso reconozca las actualizaciones y compilaciones de dependencias según lo previsto, debe usar el control de versiones semántico (x.x.x) para la imagen base y los componentes. Para obtener más información sobre el control de versiones semántico para los recursos de Generador de Imágenes, consulte Control de versiones semántico en Generador de imágenes.

5. Para continuar en el paso siguiente, seleccione Siguiente.

Paso 2: elegir receta

1. El valor predeterminado de Generador de Imágenes es Usar la receta existente en la sección Receta. Por primera vez, elija la opción Crear nueva receta.
2. En la sección Tipo de imagen, elija la opción imagen de Docker para crear una canalización de contenedores que genere una imagen de Docker y la distribuya a los repositorios de Amazon ECR de las regiones de destino.
3. En la sección General, introduzca las siguientes casillas obligatorias:

   - Nombre: el nombre de su receta

- versión: la versión de su receta (use el formato) <major>.<minor>.<patch>, donde major, minor y patch son valores enteros). Por lo general, las recetas nuevas comienzan con `1.0.0`.

4. En la sección Imagen de origen, mantenga los valores predeterminados para Seleccionar imagen, Sistema operativo (SO) de la imagen y Origen de la imagen. El resultado es una lista de imágenes de contenedor de Amazon Linux 2, administradas de Amazon, entre las que puede elegir para su imagen base.

   a. En el menú desplegable Nombre de la imagen, elija una imagen.

   b. Mantenga el valor predeterminado para las opciones de control de versiones automático (utilice la última versión del sistema operativo disponible).

   > **Note**
   >
   > Esta configuración garantiza que la canalización utilice el control de versiones semántico para la imagen base, a fin de detectar las actualizaciones de dependencias en los trabajos programados automáticamente. Para obtener más información sobre el control de versiones semántico para los recursos de Generador de Imágenes, consulte Control de versiones semántico en Generador de imágenes.

5. En la sección Componentes, debe elegir al menos un componente de compilación.

   En el panel Componentes de compilación: Amazon Linux, puede navegar por los componentes que aparecen en la página. Utilice el control de paginación situado en la esquina superior derecha para navegar por los componentes adicionales que están disponibles para su sistema operativo de imagen base. También puede buscar componentes específicos o crear su propio componente de compilación mediante el administrador de componentes.

   Para este tutorial, elija un componente que actualice Linux con las actualizaciones de seguridad más recientes, de la siguiente manera:

   a. Filtre los resultados introduciendo la palabra `update` en la barra de búsqueda que se encuentra en la parte superior del panel.

   b. Seleccione la casilla de verificación del componente de compilación `update-linux`.

   c. Desplácese hacia abajo y, en la esquina superior derecha de la lista de componentes seleccionados, elija Expandir todo.

   d. Mantenga el valor predeterminado para las opciones de control de versiones (utilice la última versión del componente disponible).

> **Note**
>
> Esta configuración garantiza que la canalización utilice el control de versiones semántico para el componente seleccionado a fin de detectar las actualizaciones de dependencias en los trabajos programados automáticamente. Para obtener más información sobre el control de versiones semántico para los recursos de Generador de Imágenes, consulte Control de versiones semántico en Generador de imágenes.

Si ha seleccionado un componente que tiene parámetros de entrada, también verá los parámetros en esta área. No se abordan los parámetros en este tutorial. Para obtener más información sobre el uso de parámetros de entrada en los componentes y su configuración en las recetas, consulte Tutorial: creación de un documento del componente con los parámetros de entrada.

Reordenar los componentes (opcional)

Si ha elegido más de un componente para incluirlos en la imagen, puede utilizar la drag-and-drop acción para reorganizarlos en el orden en que deben ejecutarse durante el proceso de creación.

> **Note**
>
> Los componentes de endurecimiento del CIS no siguen las reglas de ordenación de componentes estándar de las recetas de Generador de Imágenes. Los componentes de endurecimiento del CIS siempre se ejecutan en último lugar para garantizar que las pruebas de referencia se ejecuten con la imagen de salida.

1. Vuelva a la lista de componentes disponibles.
2. Seleccione la casilla de verificación del componente de compilación `update-linux-kernel-mainline` (o cualquier otro componente que elija).
3. Desplácese hacia abajo hasta la lista de componentes seleccionados para comprobar que haya al menos dos resultados.
4. Es posible que no se haya ampliado el control de versiones de los componentes recién agregados. Para ampliar las opciones de control de versiones, puede seleccionar la flecha

junto a las opciones de control de versiones o puede activar y desactivar la opción Expandir todo para ampliar el control de versiones de todos los componentes seleccionados.

5. Elija uno de los componentes y arrástrelo hacia arriba o hacia abajo para cambiar el orden en el que se ejecutarán los componentes.
6. Para eliminar el componente `update-linux-kernel-mainline`, seleccione `X` en la esquina superior derecha del cuadro de componentes.
7. Repita el paso anterior para eliminar cualquier otro componente que haya añadido, y deje solo el componente `update-linux` seleccionado.

6. En la sección de plantilla de Dockerfile, seleccione la opción Usar ejemplo. En el panel de contenido, observe las variables contextuales en las que Generador de Imágenes coloca información de compilación o scripts, basándose en la receta de la imagen de contenedor.

   De forma predeterminada, Generador de imágenes utiliza las siguientes variables contextuales en el Dockerfile.

   parentImage (obligatorio)

   En el momento de la compilación, esta variable se convierte en la imagen base de la receta.

   Ejemplo:

   ```
   FROM
   {{{ imagebuilder:parentImage }}}
   ```

   entornos (obligatorio si se especifican los componentes)

   Esta variable se convertirá en un script que ejecuta componentes.

   Ejemplo:

   ```
   {{{ imagebuilder:environments }}}
   ```

   componentes (opcional)

   Generador de imágenes resuelve los scripts de componentes de compilación y prueba para los componentes que incluye la receta del contenedor. Esta variable se puede colocar en cualquier parte del Dockerfile, después de la variable de entorno.

Ejemplo:

```
{{{ imagebuilder:components }}}
```

7. En la sección Repositorio de destino, especifique el nombre del repositorio de Amazon ECR que creó como requisito previo para este tutorial. Este repositorio se utiliza como configuración predeterminada para la configuración de distribución en la región donde se ejecuta la canalización (región 1).

> **Note**
>
> El repositorio de destino debe existir en Amazon ECR en todas las regiones de destino antes de la distribución.

8. Para continuar en el paso siguiente, seleccione Siguiente.

Paso 3: definir la configuración de la infraestructura (opcional)

Image Builder lanza EC2 instancias en su cuenta para personalizar las imágenes y ejecutar pruebas de validación. Los ajustes de configuración de la infraestructura especifican los detalles de la infraestructura de las instancias que se ejecutarán en su Cuenta de AWS servidor durante el proceso de creación.

En la sección Configuración de infraestructura, las opciones de configuración están predeterminadas en `Create infrastructure configuration using service defaults`. Esto crea un rol de IAM y un perfil de instancia asociado que las instancias de compilación utilizan para configurar las imágenes del contenedor. También puede crear su propia configuración de infraestructura personalizada o utilizar parámetros ya creados. Para obtener más información sobre los ajustes de configuración de la infraestructura, consulte CreateInfrastructureConfigurationla Referencia de la API de EC2 Image Builder.

En este tutorial, utilizamos la configuración predeterminada.

- Para continuar en el paso siguiente, seleccione Siguiente.

Paso 4: definir los ajustes de distribución (opcional)

La configuración de distribución consiste en las regiones de destino y el nombre del repositorio de Amazon ECR de destino. Las imágenes de Docker de salida se implementan en el repositorio de Amazon ECR denominado en cada región.

En la sección Ajustes de distribución, las opciones de configuración están predeterminadas en `Create distribution settings using service defaults`. Esta opción distribuirá la imagen de Docker de salida al repositorio de Amazon ECR especificado en la receta del contenedor para la región donde se ejecuta la canalización (región 1). Si elige `Create new distribution settings`, puede anular el repositorio de ECR de la región actual y añadir más regiones para la distribución.

En este tutorial, utilizamos la configuración predeterminada.

- Para continuar en el paso siguiente, seleccione Siguiente.

Paso 5: Revisar

La sección de revisión muestra todos los ajustes que ha configurado. Para editar la información de una sección determinada, presione el botón Editar situado en la esquina superior derecha de la sección de pasos. Por ejemplo, si quiere cambiar el nombre de la canalización, presione el botón Editar en la esquina superior derecha de la sección Paso 1: detalles de la canalización.

1. Cuando haya revisado la configuración, seleccione Crear canalización para crear su canalización.
2. Puede ver los mensajes de éxito o falla en la parte superior de la página, a medida que se crean los recursos para los ajustes de la distribución, la configuración de la infraestructura, la nueva receta y la canalización. Para ver los detalles de un recurso, incluido el identificador del recurso, seleccione Ver detalles.
3. Después de ver los detalles de un recurso, puede ver los detalles de otros recursos seleccionando el tipo de recurso en el panel de navegación. Por ejemplo, para ver los detalles de su nueva canalización, seleccione Canalizaciones de imágenes en el panel de navegación. Si la compilación se realizó correctamente, la nueva canalización se mostrará en la lista de canalizaciones de imágenes.

Paso 6: limpiar

Su entorno Generador de Imágenes, al igual que su hogar, necesita un mantenimiento habitual para ayudarlo a encontrar lo que necesita y completar sus tareas sin tener que preocuparse por el desorden. Asegúrese de limpiar periódicamente los recursos temporales que creó para las pruebas. De lo contrario, es posible que se olvide de esos recursos y, más adelante, no recuerde para qué se utilizaron. Para entonces, es posible que no esté claro si puede deshacerse de ellos de manera segura.

> **Tip**
>
> Para evitar errores de dependencia al eliminar recursos, asegúrese de eliminarlos en el siguiente orden:
>
> 1. Canalización de imágenes
> 2. Receta de imagen
> 3. Todos los recursos restantes

Siga los pasos de esta sección para eliminar los recursos que ha creado en este tutorial:

Eliminación de la canalización

1. Para ver una lista de las canalizaciones de compilación creadas en su cuenta, seleccione las canalizaciones de imágenes en el panel de navegación.
2. Seleccione la casilla de verificación junto al nombre de canalización para seleccionar la canalización que desea eliminar.
3. En la parte superior del panel Canalizaciones de imágenes, en el menú Acciones, seleccione Eliminar.
4. Para confirmar la eliminación, ingrese `Delete` en el cuadro y seleccione Eliminar.

Eliminación de la receta de contenedor

1. Para ver una lista de las recetas de contenedor creadas en su cuenta, seleccione Recetas de contenedor en el panel de navegación.
2. Seleccione la casilla de verificación junto al nombre de la receta para seleccionar la receta que desea eliminar.

3. En la parte superior del panel Recetas de contenedor, en el menú Acciones, seleccione Eliminar receta.
4. Para confirmar la eliminación, ingrese `Delete` en el cuadro y seleccione Eliminar.

Eliminación de configuración de infraestructura

1. Para ver una lista de las configuraciones de infraestructura creadas en su cuenta, seleccione Configuración de infraestructura en el panel de navegación.
2. Seleccione la casilla de verificación junto al nombre de la configuración para seleccionar la configuración de infraestructura que desea eliminar.
3. En la parte superior del panel Configuraciones de infraestructura, seleccione Eliminar.
4. Para confirmar la eliminación, ingrese `Delete` en el cuadro y seleccione Eliminar.

Eliminación de ajustes de distribución

1. Para ver una lista de los ajustes de distribución creados en su cuenta, seleccione Ajustes de distribución en el panel de navegación.
2. Seleccione la casilla de verificación junto al nombre de la configuración para seleccionar los ajustes de distribución que creó para este tutorial.
3. En la parte superior del panel Ajustes de distribución, seleccione Eliminar.
4. Para confirmar la eliminación, ingrese `Delete` en el cuadro y seleccione Eliminar.

Eliminación de la imagen

Siga estos pasos para comprobar que ha eliminado cualquier imagen que se haya creado de la canalización del tutorial. No es probable que este tutorial cree una imagen a menos que haya transcurrido suficiente tiempo desde que creó la canalización para que se ejecute, de acuerdo con el cronograma de compilación.

1. Para ver una lista de las imágenes creadas en su cuenta, seleccione Imágenes en el panel de navegación.
2. Elija la versión de la imagen que desea eliminar. Esto abre la página Versiones de compilación de imágenes.
3. Seleccione la casilla de verificación junto a la versión de cualquier imagen que desea eliminar. Puede seleccionar más de una versión de imagen a la vez.

4. En la parte superior del panel Versiones de compilación de imágenes, seleccione Eliminar versión.
5. Para confirmar la eliminación, ingrese `Delete` en el cuadro y seleccione Eliminar.

Tutorial: creación de un documento del componente con los parámetros de entrada

Puede administrar los componentes de Image Builder, incluida la creación y configuración de los parámetros de los componentes, directamente desde la consola de EC2 Image Builder AWS CLI, desde o desde la API Image Builder o SDKs. En esta sección, abordaremos la creación y el uso de parámetros en el componente y la configuración de los parámetros de los componentes mediante la consola y los AWS CLI comandos de Image Builder en tiempo de ejecución.

> **Important**
>
> Los parámetros del componente son valores de texto sin formato y se registran en AWS CloudTrail. Le recomendamos que utilice AWS Secrets Manager o el almacén de AWS Systems Manager parámetros para almacenar sus secretos. Para obtener más información sobre Secrets Manager, consulte ¿Qué es Secrets Manager? en la Guía del usuario de AWS Secrets Manager . Para obtener más información acerca del almacén de parámetros de AWS Systems Manager , consulte Almacén de parámetros de AWS Systems Manager en la Guía del usuario de AWS Systems Manager .

Uso de los parámetros en su documento del componente YAML

Para crear un componente, debe proporcionar un documento del componente de la aplicación YAML o JSON. El documento contiene el código que se ejecuta durante las fases y los pasos que defina para personalizar la imagen. La receta que hace referencia al componente puede establecer los parámetros para personalizar los valores en el tiempo de ejecución, con valores predeterminados que surten efecto si el parámetro no está establecido en un valor específico.

Creación de un documento del componente con los parámetros de entrada

Esta sección muestra cómo definir y usar los parámetros de entrada en el documento del componente YAML.

Para crear un documento del componente de la aplicación YAML que utilice parámetros y ejecute comandos en las instancias de compilación o prueba de Generador de Imágenes, siga los pasos que coincidan con el sistema operativo de su imagen:

Linux

Creación de un documento del componente YAML

Utilice una herramienta de edición de archivos para crear un archivo de documento de componente. Los ejemplos de documentación utilizan un archivo denominado *hello-world-test.yaml* que incluye el siguiente contenido:

```
# Document Start
#
name: "HelloWorldTestingDocument-Linux"
description: "Hello world document to demonstrate parameters."
schemaVersion: 1.0
parameters:
  - MyInputParameter:
      type: string
      default: "It's me!"
      description: This is an input parameter.
phases:
  - name: build
    steps:
      - name: HelloWorldStep
        action: ExecuteBash
        inputs:
          commands:
            - echo "Hello World! Build phase. My input parameter value is
 {{ MyInputParameter }}"

  - name: validate
    steps:
      - name: HelloWorldStep
        action: ExecuteBash
        inputs:
          commands:
            - echo "Hello World! Validate phase. My input parameter value is
 {{ MyInputParameter }}"

  - name: test
    steps:
```

```
      - name: HelloWorldStep
        action: ExecuteBash
        inputs:
          commands:
            - echo "Hello World! Test phase. My input parameter value is
 {{ MyInputParameter }}"
# Document End
```

> **Tip**
>
> Utilice una herramienta como este Validador YAML en línea o una extensión YAML lint en su entorno de código para verificar que su YAML está bien formado.

Windows

Creación de un documento del componente YAML

Utilice una herramienta de edición de archivos para crear un archivo de documento de componente. Los ejemplos de documentación utilizan un archivo denominado *hello-world-test.yaml* que incluye el siguiente contenido:

```
# Document Start
#
name: "HelloWorldTestingDocument-Windows"
description: "Hello world document to demonstrate parameters."
schemaVersion: 1.0
parameters:
  - MyInputParameter:
      type: string
      default: "It's me!"
      description: This is an input parameter.
phases:
  - name: build
    steps:
      - name: HelloWorldStep
        action: ExecutePowerShell
        inputs:
          commands:
            - Write-Host "Hello World! Build phase. My input parameter value is
 {{ MyInputParameter }}"
```

```
  - name: validate
    steps:
      - name: HelloWorldStep
        action: ExecutePowerShell
        inputs:
          commands:
            - Write-Host "Hello World! Validate phase. My input parameter value is
 {{ MyInputParameter }}"

  - name: test
    steps:
      - name: HelloWorldStep
        action: ExecutePowerShell
        inputs:
          commands:
            - Write-Host "Hello World! Test phase. My input parameter value is
 {{ MyInputParameter }}"
# Document End
```

> **Tip**
>
> Utilice una herramienta como este Validador YAML en línea o una extensión YAML lint en su entorno de código para verificar que su YAML está bien formado.

macOS

Creación de un documento del componente YAML

Utilice una herramienta de edición de archivos para crear un archivo de documento de componente. Los ejemplos de documentación utilizan un archivo denominado *`hello-world-test.yaml`* que incluye el siguiente contenido:

```
# Document Start
#
name: "HelloWorldTestingDocument-macOS"
description: "Hello world document to demonstrate parameters."
schemaVersion: 1.0
parameters:
  - MyInputParameter:
      type: string
      default: "It's me!"
```

```
      description: This is an input parameter.
phases:
  - name: build
    steps:
      - name: HelloWorldStep
        action: ExecuteBash
        inputs:
          commands:
            - echo "Hello World! Build phase. My input parameter value is
 {{ MyInputParameter }}"

  - name: validate
    steps:
      - name: HelloWorldStep
        action: ExecuteBash
        inputs:
          commands:
            - echo "Hello World! Validate phase. My input parameter value is
 {{ MyInputParameter }}"

  - name: test
    steps:
      - name: HelloWorldStep
        action: ExecuteBash
        inputs:
          commands:
            - echo "Hello World! Test phase. My input parameter value is
 {{ MyInputParameter }}"
# Document End
```

> **Tip**
>
> Utilice una herramienta como este Validador YAML en línea o una extensión YAML lint en su entorno de código para verificar que su YAML está bien formado.

Para obtener más información sobre las fases, los pasos y la sintaxis de los documentos del componente de la aplicación TOE de AWS , consulte Cómo usar documentos en TOE de AWS. Para obtener más información sobre los parámetros y sus requisitos, consulte la sección Parameters de la página Definir y referenciar variables en TOE de AWS.

Creación de un componente a partir del documento de componente de YAML

Sea cual sea el método que utilices para crear un TOE de AWS componente, el documento del componente de la aplicación YAML siempre es obligatorio como referencia.

- Para crear un componente directamente desde su documento YAML con la consola de Generador de imágenes, consulte Creación de un componente personalizado de la consola.
- Para crear un componente desde la línea de comandos con el comando create-component de Generador de imágenes, consulte Cree un componente personalizado a partir del AWS CLI. Reemplace el nombre del documento YAML de esos ejemplos por el nombre del documento YAML de Hello World (*`hello-world-test.yaml`*).

Establecimiento de los parámetros del componente en una receta de Generador de imágenes desde la consola

La configuración de los parámetros del componente funciona de la misma manera para las recetas de imágenes y las recetas de contenedores. Al crear una receta nueva o una nueva versión de una receta, se eligen los componentes que se van a incluir en las listas Componentes de compilación y Componentes de prueba. Las listas de componentes incluyen los componentes que son aplicables al sistema operativo base que haya elegido para la imagen.

Después de seleccionar un componente, se muestra en la sección Componentes seleccionados, justo debajo de las listas de componentes. Se muestran las opciones de configuración para cada componente seleccionado. Si el componente tiene parámetros de entrada definidos, se muestran como una sección ampliable denominada Parámetros de entrada.

Se muestran las siguientes configuraciones de parámetros para cada parámetro definido para el componente:

- Nombre del parámetro (no editable): el nombre del parámetro.
- Descripción (no editable): la descripción del parámetro.
- Tipo (no editable): el tipo de datos del valor del parámetro.
- Valor: el valor del parámetro. Si utiliza este componente por primera vez en esta receta y se definió un valor predeterminado para el parámetro de entrada, el valor predeterminado aparece en el cuadro Valor con el texto atenuado. Si no se especifica ningún otro valor, Generador de imágenes utiliza el valor predeterminado.

Utilice un parámetro de imagen base en su receta

Al crear una receta para la personalización de imágenes, hay varias formas de identificar la imagen base con la que se comienza. Si especificas el ID de Amazon Machine Image (AMI) para tu imagen base y esa imagen base se actualiza, es posible que su ID de AMI cambie y tengas que actualizar la receta para que coincida.

En lugar de cambiar la receta cada vez que cambia el ID de la imagen base, puede definir un AWS Systems Manager parámetro de almacén de parámetros (parámetro SSM) para almacenar el valor del ID de AMI de la imagen base y, a continuación, utilizar el parámetro para especificar la imagen base en la receta. En el AWS caso de la versión gestionada AMIs, puede utilizar un parámetro público para la versión más reciente.

En este tutorial, se explica el proceso de creación de un parámetro de ID de AMI y su uso en una receta de imagen. Los pasos de Image Builder de este tutorial se basan en la consola.

Contenido

- Paso 1: Busque o cree un parámetro del almacén de parámetros
- Paso 2: Configurar los permisos de IAM (opcional)
- Paso 3: Cree una receta de imagen que utilice el parámetro

Paso 1: Busque o cree un parámetro del almacén de parámetros

El proceso de este paso depende del tipo de AMI que especifique para la imagen base. AMIsEn AWS el caso de la administración, puede utilizar un parámetro público que haga referencia a la versión actual. Es posible que algunos parámetros no estén disponibles en todos Regiones de AWS.

Para empezar, abra la pestaña correspondiente a su AMI.

AWS managed AMI

Si la imagen base es una AMI AWS gestionada, puede utilizar parámetros públicos para especificar el ID de la AMI, en lugar de crear su propio parámetro. Para encontrar el parámetro público de su AMI, consulte Descubrimiento de parámetros públicos en la Guía del AWS Systems Manager usuario.

Custom AMI

Para crear un parámetro de ID de AMI, siga las instrucciones para crear parámetros de almacén de parámetros en Systems Manager con la consola AWS CLI, o PowerShell. Proporcione los siguientes valores para asegurarse de que el valor del parámetro es un ID de AMI.

Nivel de parámetros: `Standard`

Tipo: `String`

Tipo de datos: seleccione`aws:ec2:image`. Al especificar este tipo, el sistema valida el valor introducido para garantizar que se trata de un ID de AMI.

Valor: introduzca un ID de AMI válido (por ejemplo,***ami-1234567890abcdef1***).

Paso 2: Configurar los permisos de IAM (opcional)

Para utilizar un parámetro del almacén de parámetros de Systems Manager (parámetro SSM), ya sea público o privado, se deben especificar las siguientes acciones del almacén de parámetros de Systems Manager en una función de IAM, con el parámetro listado como recurso. El rol vinculado al servicio Image Builder concede permiso para obtener parámetros públicos o para obtener o actualizar parámetros privados que tienen un `/imagebuilder/` prefijo. Para los parámetros privados que no tienen ese prefijo, puede añadir permisos a su función de ejecución.

- `ssm:GetParameter`— Esta acción te permite usar un parámetro SSM para especificar la imagen base de tu receta.
- `ssm:PutParameter`— Esta acción le permite almacenar el ID de AMI de salida en un parámetro SSM durante la distribución. La definición de política tiene el mismo aspecto, pero este tutorial no incluye la acción de colocación en la política de ejemplo.

1. Crea un rol personalizado (opcional)

   Al crear una canalización o utilizar el comando create-image en AWS CLI, solo puede especificar una función de ejecución de Image Builder. Si ha definido una función de ejecución del flujo de trabajo de Image Builder, añadiría cualquier permiso de función adicional a esa función. De lo contrario, crearía un nuevo rol personalizado que incluya los permisos necesarios. Si ya ha definido un rol de ejecución personalizado, puede omitir este paso.

Siga el proceso de creación de un rol para delegar permisos a un AWS servicio en la Guía del AWS Identity and Access Management usuario.

2. Añada permisos a su función personalizada

   Para añadir los permisos de los parámetros SSM a su rol personalizado, siga el proceso de actualización de la política de permisos para un rol de la Guía del AWS Identity and Access Management usuario.

   En el siguiente ejemplo de política, se muestra la `ssm:GetParameter` acción con un parámetro que se creó en tu cuenta.

   JSON

```
{
 "Version":"2012-10-17",
 "Statement": [
  {
   "Sid": "PrivateParameterCustomRole",
   "Effect": "Allow",
   "Action": "ssm:GetParameter",
   "Resource": "arn:aws:ssm:*:111122223333:parameter/parameter-name"
  }
 ]
}
```

Para obtener más información sobre los recursos de parámetros públicos, consulte Llamar a los parámetros públicos de la AMI en la Guía del AWS Systems Manager usuario.

Paso 3: Cree una receta de imagen que utilice el parámetro

1. Abra la consola EC2 de Image Builder en https://console.aws.amazon.com/imagebuilder/.
2. Seleccione Recetas con imágenes y, a continuación, seleccione Crear receta con imágenes en la página de la lista.
3. Rellene la sección Imagen base de la siguiente manera:

   a. Elija la opción Usar AMI personalizada. Muestra campos adicionales en los que puede introducir el ID de AMI o un parámetro de SSM que contenga el ID de AMI.

b. Elija la opción de parámetro SSM.

c. En el campo de parámetro SSM, introduzca el nombre del parámetro o el nombre de recurso de Amazon (ARN) del parámetro que creó en el paso 1. Si introduce el nombre, no tendrá el prefijo en la consola.

4. Complete el resto de la configuración de la receta según sea necesario.

> **Note**
>
> Si configura la imagen principal a través de otras interfaces, como la AWS CLI, el nombre del parámetro debe tener un prefijo de `ssm:` (por ejemplo,`ssm:/ImageBuilder-Tutorial/BaseAMI`.

Uso de componentes para personalizar su imagen de Generador de imágenes

Image Builder utiliza la aplicación de administración de componentes Ejecutor y orquestador de tareas de AWS(TOE de AWS) para organizar flujos de trabajo complejos. Los componentes de compilación y prueba que funcionan con la aplicación TOE de AWS se basan en documentos YAML que definen los scripts para personalizar o probar su imagen. Para las imágenes de AMI, Image Builder instala los componentes y la aplicación de administración de TOE de AWS componentes en sus instancias de EC2 compilación y prueba de Amazon. En el caso de las imágenes de contenedores, los TOE de AWS componentes y la aplicación de administración de componentes se instalan dentro del contenedor en ejecución.

Image Builder se utiliza TOE de AWS para realizar todas las actividades en la instancia. No se requiere ninguna configuración adicional con la que interactuar TOE de AWS al ejecutar comandos de Image Builder o utilizar la consola de Image Builder.

> **Note**
>
> Cuando un componente administrado por Amazon llega al final de su vida útil de soporte, deja de recibir mantenimiento. Aproximadamente cuatro semanas antes de que esto ocurra, todas las cuentas que utilicen el componente recibirán una notificación de parte de su AWS Health Dashboard y una lista de las recetas afectadas de su cuenta. Para obtener más información AWS Health, consulte la Guía AWS Health del usuario.

Etapas del flujo de trabajo para compilar una nueva imagen

El flujo de trabajo de Generador de Imágenes para compilar nuevas imágenes incluye las siguientes dos etapas distintas.

1. Etapa de compilación (previa a la instantánea): durante la etapa de compilación, realizas cambios en la instancia de EC2 compilación de Amazon en la que se ejecuta tu imagen base para crear la línea base de la nueva imagen. Por ejemplo, la receta puede incluir componentes que instalen una aplicación o modifiquen la configuración del firewall del sistema operativo.

   Las siguientes fases de su documento de componente se ejecutan durante la etapa de compilación:

- build
- validar

Una vez que esta etapa se complete correctamente, Generador de Imágenes crea una instantánea o imagen de contenedor que utilizará para la etapa de prueba y posteriores.

2. Etapa de prueba (posterior a la instantánea): durante la etapa de prueba, existen algunas diferencias entre las imágenes que crean imágenes AMIs y las que contienen imágenes. Para los flujos de trabajo de AMI, Image Builder lanza una EC2 instancia a partir de la instantánea que creó como paso final de la fase de creación. Las pruebas se ejecutan en la nueva instancia para validar la configuración y garantizar que la instancia funcione según lo previsto. En el caso de los flujos de trabajo de contenedor, las pruebas se ejecutan en la misma instancia que se utilizó para la creación.

   La siguiente fase de su documento de componentes se ejecuta para cada componente incluido en la receta durante la etapa de prueba de creación de la imagen:

   - prueba

   Esta fase de los componentes se aplica a los tipos de componentes de compilación y prueba. Una vez que esta etapa se complete correctamente, Generador de Imágenes podrá crear y distribuir su imagen final a partir de la instantánea o la imagen de contenedor.

> **Note**
>
> Si bien el marco de la TOE de AWS aplicación permite definir muchas fases en un documento de componentes, Image Builder tiene reglas estrictas sobre qué fases ejecuta y durante qué etapas las ejecuta. Para que un componente se ejecute durante la fase de compilación de imágenes, el documento del componente debe definir al menos una de estas fases: `build` o `validate`. Para que un componente se ejecute durante la etapa de prueba de imágenes, el documento del componente debe definir la fase `test` y no otras fases. Como Generador de Imágenes ejecuta las etapas de forma independiente, el encadenamiento de referencias en los documentos de los componentes no puede cruzar los límites de las etapas. No puede encadenar un valor de una fase que se ejecuta en la etapa de compilación a una fase que se ejecuta en la etapa de prueba. Sin embargo, puede definir los parámetros de entrada para el objetivo deseado y transferir los valores a través de la línea de comandos. Para obtener más información sobre cómo configurar los parámetros de

los componentes en sus recetas de Generador de Imágenes, consulte Tutorial: creación de un documento del componente con los parámetros de entrada.

Para ayudar a solucionar problemas en la instancia de compilación o prueba,TOE de AWS crea una carpeta de registro que contiene el documento de entrada y los archivos de registro para realizar un seguimiento de lo que ocurre cada vez que se ejecuta un componente. Si configuró un bucket de Amazon S3 en la configuración de su canalización, los registros también se escriben allí. Para obtener más información acerca de los documentos YAML y la salida de registros, consulte Utilice el marco de documentos de TOE de AWS componentes para componentes personalizados.

> **Tip**
>
> Cuando tiene muchos componentes de los que realizar un seguimiento, el etiquetado le ayuda a identificar un componente o versión específicos en función de las etiquetas que les haya asignado. Para obtener más información sobre cómo etiquetar los recursos mediante los comandos de Image Builder de AWS CLI, consulte la Etiquetar recursos sección de esta guía.

En esta sección se explica cómo enumerar, ver, crear e importar componentes mediante la consola de Generador de Imágenes o los comandos de la AWS CLI.

Temas

- Enumeración y visualización de los detalles de los componentes
- Utilice AWS Marketplace componentes para personalizar la imagen
- Utilice componentes administrados para personalizar su imagen de Generador de imágenes
- Desarrollo de componentes personalizados para su imagen de Generador de imágenes
- Cómo utiliza Image Builder la Ejecutor y orquestador de tareas de AWS aplicación para gestionar los componentes

Enumeración y visualización de los detalles de los componentes

En esta sección se describe cómo encontrar información y ver los detalles de los componentes que se utilizan en las recetas de EC2 Image Builder.

Detalles de los componentes

- Enumeración de componentes de Generador de imágenes
- Enumere las versiones de compilación de los componentes del AWS CLI
- Obtenga los detalles de los componentes en AWS CLI
- Obtenga los detalles de la política de componentes en el AWS CLI

Enumeración de componentes de Generador de imágenes

Puede utilizar uno de los métodos siguientes para enumerar y filtrar componentes de Generador de imágenes.

Consola de administración de AWS

Para mostrar una lista de los componentes del Consola de administración de AWS, siga estos pasos:

1. Abra la consola EC2 de Image Builder en https://console.aws.amazon.com/imagebuilder/.
2. En el panel de navegación, seleccione Componentes. De forma predeterminada, Generador de Imágenes muestra una lista de los componentes que pertenecen a su cuenta.
3. Si lo desea, puede filtrar según la propiedad de los componentes. Para ver los componentes que no son de su propiedad, pero a los que tiene acceso, amplíe la lista desplegable de tipos de propietario y seleccione uno de los valores. La lista de tipos de propietario se encuentra en la barra de búsqueda, junto al cuadro de texto de búsqueda. Puede seleccionar de entre los siguientes valores:
   - AWS Marketplace— Componentes que están asociados directamente a la suscripción de un AWS Marketplace producto.
   - Inicio rápido (administrado por Amazon): componentes disponibles públicamente que Amazon crea y mantiene.
   - De mi propiedad: componentes que usted creó. Esta es la selección predeterminada.
   - Compartido conmigo: componentes que otros han creado y compartido contigo desde su cuenta.
   - Administrados por terceros: componentes que son propiedad de un tercero y a los que usted se ha suscrito. AWS Marketplace

AWS CLI

En el siguiente ejemplo, se muestra cómo utilizar el comando de list-components para devolver una lista de los componentes de Generador de imágenes que son propiedad de su cuenta.

```
aws imagebuilder list-components
```

Si lo desea, puede filtrar según la propiedad de los componentes. El atributo de propietario define quién es el propietario de los componentes que desea enumerar. De forma predeterminada, esta solicitud devuelve una lista de los componentes que pertenecen a su cuenta. Para filtrar los resultados por propietario del componente, especifique uno de los siguientes valores con el parámetro `--owner` cuando ejecute el comando list-components.

Valores del propietario del componente

- `AWSMarketplace`
- `Amazon`
- `Self`
- `Shared`
- `ThirdParty`

Los ejemplos siguientes muestran el comando list-components con el parámetro `--owner` para filtrar los resultados.

```
aws imagebuilder list-components --owner Self
{
    "requestId": "012a3456-b789-01cd-e234-fa5678b9012b",
    "componentVersionList": [
        {
            "arn": "arn:aws:imagebuilder:us-west-2:123456789012:component/sample-
component01/1.0.0",
            "name": "sample-component01",
            "version": "1.0.0",
            "platform": "Linux",
            "type": "BUILD",
            "owner": "123456789012",
            "dateCreated": "2020-09-24T16:58:24.444Z"
        },
```

```
        {
            "arn": "arn:aws:imagebuilder:us-west-2:123456789012:component/sample-
component01/1.0.1",
            "name": "sample-component01",
            "version": "1.0.1",
            "platform": "Linux",
            "type": "BUILD",
            "owner": "123456789012",
            "dateCreated": "2021-07-10T03:38:46.091Z"
        }
    ]
}
```

```
aws imagebuilder list-components --owner Amazon
```

```
aws imagebuilder list-components --owner Shared
```

```
aws imagebuilder list-components --owner ThirdParty
```

Enumere las versiones de compilación de los componentes del AWS CLI

En el ejemplo siguiente se muestra cómo utilizar el comando list-component-build-versions para enumerar las versiones de creación de componentes que tienen una versión semántica específica. Para obtener más información sobre el control de versiones semántico para los recursos de Generador de Imágenes, consulte Control de versiones semántico en Generador de imágenes.

```
aws imagebuilder list-component-build-versions --component-version-arn
 arn:aws:imagebuilder:us-west-2:123456789012:component/example-component/1.0.1
{
    "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "componentSummaryList": [
        {
            "arn": "arn:aws:imagebuilder:us-west-2:123456789012:component/
examplecomponent/1.0.1/1",
            "name": "examplecomponent",
            "version": "1.0.1",
            "platform": "Linux",
            "type": "BUILD",
            "owner": "123456789012",
```

```
            "description": "An example component that builds, validates and tests an
 image",
            "changeDescription": "Updated version.",
            "dateCreated": "2020-02-19T18:53:45.940Z",
            "tags": {
                "KeyName": "KeyValue"
            }
        }
    ]
}
```

Obtenga los detalles de los componentes en AWS CLI

El siguiente ejemplo muestra cómo utilizar el comando get-component para obtener los detalles del componente al especificar el nombre de recurso de Amazon (ARN) del componente.

```
aws imagebuilder get-component --component-build-version-arn arn:aws:imagebuilder:us-
west-2:123456789012:component/example-component/1.0.1/1
   {
    "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11112",
    "component": {
        "arn": "arn:aws:imagebuilder:us-west-2:123456789012:component/
examplecomponent/1.0.1/1",
        "name": "examplecomponent",
        "version": "1.0.1",
        "type": "BUILD",
        "platform": "Linux",
        "owner": "123456789012",
        "data": "name: HelloWorldTestingDocument\ndescription: This is hello world
 testing document... etc.\"\n",
        "encrypted": true,
        "dateCreated": "2020-09-24T16:58:24.444Z",
        "tags": {}
    }
}
```

Obtenga los detalles de la política de componentes en el AWS CLI

En el ejemplo siguiente se muestra cómo utilizar el comando get-component-policy para obtener detalles de una política de componentes al especificar el ARN del componente.

```
aws imagebuilder get-component-policy --component-arn arn:aws:imagebuilder:us-
west-2:123456789012:component/example-component/1.0.1
```

Utilice AWS Marketplace componentes para personalizar la imagen

Además de una amplia selección de imágenes creadas por proveedores de software independientes (ISVs),AWS Marketplace ofrece componentes que puede utilizar para personalizar sus propias imágenes de Image Builder. Debe suscribirse a estos AWS Marketplace componentes antes de poder utilizarlos en su receta de imagen para crear una nueva imagen.

Al especificar un AWS Marketplace componente en una receta de imagen, Image Builder valida la suscripción y realiza comprobaciones de dependencias para garantizar que dispone de los recursos que necesita para utilizarla. Cuando la validación se realiza correctamente, Image Builder crea descargas seguras para el componente y sus artefactos para que las utilicen las compilaciones de Image Pipeline.

Descubra los componentes AWS Marketplace

Puede encontrar los componentes de AWS Marketplace software que puede utilizar en sus recetas en la página Descubra productos de la consola Image Builder, de la siguiente manera.

1. Abra la consola EC2 de Image Builder en https://console.aws.amazon.com/imagebuilder/.
2. En el panel de navegación, selecciona Descubrir productos en la AWS Marketplacesección.
3. Seleccione la pestaña Components (Componentes). En esta pestaña se muestran todos los AWS Marketplace productos que utilizan la opción de entrega que incluye los componentes asociados AWS Marketplace.
4. Para buscar productos de software específicos que incluyan componentes, puede introducir parte del nombre en la barra de búsqueda o filtrar por `StatusOperating System`,`Publisher`, o`Categories`. La barra de búsqueda también contiene controles de paginación para los resultados.

Resultados

Cada AWS Marketplace producto tiene su propio panel de detalles que incluye la siguiente información.

El nombre y el logotipo del AWS Marketplace producto

El nombre del producto de software está vinculado a los detalles del producto en AWS Marketplace. Puede seleccionar el enlace para obtener más información sobre el producto en AWS Marketplace. Como alternativa, puedes ver un resumen de las opciones de suscripción y suscribirte directamente desde los resultados de búsqueda con el botón Ver opciones de suscripción si ya has investigado.

Versión

Contiene la versión principal del componente.

Sistema operativo

El sistema operativo para el que está diseñado el componente.

Publicador

El editor del componente. Está vinculado a la página de detalles del editor en AWS Marketplace. Se abrirá la página de detalles del publicador en una pestaña nueva en su navegador.

Categorías

Una o más categorías de AWS Marketplace productos que se aplican al componente.

Estado

Muestra si está suscrito a este producto. Si no está suscrito, puede seleccionar Ver opciones de suscripción para ver un resumen de las opciones de suscripción del AWS Marketplace producto y, si lo desea, suscribirse directamente desde la consola de Image Builder.

Componentes asociados

Si el AWS Marketplace producto tiene una o más versiones incluidas en la suscripción, se muestran en la sección Componentes asociados. La sección se contrae inicialmente y muestra un recuento de los componentes asociados. Puede ampliar la sección para ver más detalles.

> **Note**
>
> El componente Center for Internet Security (CIS) asociado a su producto de AWS Marketplace imagen no aparece en los resultados de Discover Products. Si te suscribes a su producto de imagen, el componente asociado se muestra en la página de suscripciones y, como componente de terceros, en el cuadro de diálogo Crear receta de imagen. Para

obtener más información sobre el componente, consulteComponentes de endurecimiento de CIS.

Suscríbase a AWS Marketplace los componentes

Cuando hayas encontrado un AWS Marketplace producto con componentes que desees usar en tus recetas, puedes suscribirte a él directamente desde la consola de Image Builder, de la siguiente manera, o puedes suscribirte desde la AWS Marketplace consola.

1. Abra la consola EC2 de Image Builder en https://console.aws.amazon.com/imagebuilder/.
2. En el panel de navegación, selecciona Descubrir productos en la AWS Marketplacesección.
3. Seleccione la pestaña Components (Componentes). En esta pestaña se muestran todos los AWS Marketplace productos que utilizan la opción de entrega que incluye los componentes asociados AWS Marketplace.
4. Para buscar un AWS Marketplace producto específico, introduce parte del nombre en la barra de búsqueda. Si conoces al editor, pero no sabes el nombre exacto del producto ni cómo se escribe, también puedes filtrar por ellos `Publisher` para obtener una lista de los productos que el editor tiene disponibles.
5. Selecciona el producto al que quieres suscribirte en la lista de resultados y selecciona Ver opciones de suscripción. Aquí se muestra un resumen de las opciones de suscripción del AWS Marketplace producto.
6. Seleccione Suscribirse para suscribirse al producto sin salir de la consola de Image Builder. Se le notificará que la suscripción se está procesando. Una vez suscrito, el estado se actualiza a`Subscribed`.

Para obtener más información sobre los AWS Marketplace productos a los que está suscrito actualmente, consulte los pasos de la consola que se describen en. AWS Marketplace suscripciones en Image Builder

Uso de un AWS Marketplace componente en una receta de imagen de Image Builder

Puede utilizar AWS Marketplace los componentes en las recetas de imágenes de Image Builder del mismo modo que utiliza otros tipos de componentes. Para la mayoría de los componentes asociados

a un producto de AWS Marketplace imagen, la categoría de propiedad es`AWS Marketplace`. Por ejemplo, para usar un componente de compilación de un AWS Marketplace producto al que esté suscrito, elija Agregar componentes de compilación y selecciónelo `AWS Marketplace` de la lista. Esto abre un panel de selección en la parte derecha de la interfaz de la consola que muestra AWS Marketplace los componentes.

> **Note**
>
> Si busca el componente de endurecimiento CIS`Third party managed`, seleccione uno de la lista de propietarios en lugar de`AWS Marketplace`.

Para obtener más información sobre cómo seleccionar, organizar y configurar los parámetros de los componentes, consulteCreación de una nueva versión de una receta de imagen.

Utilice componentes administrados para personalizar su imagen de Generador de imágenes

Los componentes gestionados los crea una organización externa AWS, a veces en colaboración con ella, como el Center for Internet Security (CIS), por ejemplo. Cuando utilizas componentes gestionados en tus recetas de imagen o contenedor, Amazon proporciona las versiones más recientes de los componentes a las que se han aplicado parches y otras actualizaciones. Para obtener una lista de componentes o información sobre los componentes, consulteEnumeración y visualización de los detalles de los componentes.

La siguiente lista de componentes AWS gestionados destacados incluye un componente que puede utilizar al suscribirse a CIS Hardened a AMIs través del AWS Marketplace.

Componentes destacados

- Instalación de una aplicación de componentes administrada por un paquete distribuidor para imágenes de Windows de Generador de imágenes
- Componentes de endurecimiento de CIS
- Componentes de endurecimiento de STIG administrados por Amazon para Generador de imágenes

Instalación de una aplicación de componentes administrada por un paquete distribuidor para imágenes de Windows de Generador de imágenes

AWS Systems Manager Distributor le ayuda a empaquetar y publicar software en los nodos AWS Systems Manager gestionados. Puede empaquetar y publicar su propio software o utilizar Distributor para buscar y publicar paquetes de software de agente proporcionados por AWS. Para obtener más información acerca de Systems Manager Distributor, consulte AWS Systems Manager Distributor en la Guía del usuario de AWS Systems Manager.

Componentes administrados por Distributor

Los siguientes componentes gestionados por Image Builder utilizan AWS Systems Manager Distributor para instalar paquetes de aplicaciones en instancias de Windows.

- El componente administrado `distributor-package-windows` utiliza el Distribuidor de AWS Systems Manager para instalar los paquetes de aplicaciones que especifique en la instancia de creación de imágenes de Windows. Para configurar los parámetros al incluir este componente en la receta, consulte Configuración de `distributor-package-windows` como componente independiente.
- El `aws-vss-components-windows` componente usa AWS Systems Manager Distributor para instalar el `AwsVssComponents` paquete en la instancia de compilación de imágenes de Windows. Para configurar los parámetros al incluir este componente en la receta, consulte Configuración de `aws-vss-components-windows` como componente independiente.

Para obtener más información sobre cómo utilizar los componentes administrados en su receta de Generador de Imágenes, consulte Creación de una nueva versión de una receta de imagen para recetas de imagen o Creación de una nueva versión de una receta de contenedor para recetas de contenedor. Para obtener más información sobre el `AwsVssComponents` paquete, consulte Crear una instantánea de VSS coherente con las aplicaciones en la Guía del usuario de Amazon EC2 .

Requisitos previos

Antes de utilizar los componentes de Generador de imágenes que dependen de Systems Manager Distributor para instalar paquetes de aplicaciones, debe asegurarse de que se cumplen los siguientes requisitos previos.

- Los componentes de Generador de Imágenes que utilizan Systems Manager Distributor para instalar paquetes de aplicaciones en su instancia necesitan permiso para llamar a la API de

Systems Manager. Antes de utilizar los componentes de una receta de Generador de Imágenes, debe crear el rol y la política de IAM que otorgan el permiso. Para configurar los permisos, consulte Configuración de permisos para Systems Manager Distributor.

> **Note**
>
> Por el momento, Generador de Imágenes no admite los paquetes de Systems Manager Distributor que reinician la instancia. Por ejemplo, los paquetes `AWSNVMe`, `AWSPVDrivers` y `AwsEnaNetworkDriver` de Distributor reinician la instancia y, por lo tanto, no están permitidos.

Configuración de permisos para Systems Manager Distributor

El componente `distributor-package-windows` y otros componentes que lo utilizan, por ejemplo `aws-vss-components-windows`, requieren un permiso adicional en la instancia de compilación para ejecutarse. La instancia de compilación debe poder llamar a la API de Systems Manager para iniciar la instalación de Distributor y sondear el resultado.

Siga estos procedimientos Consola de administración de AWS para crear una política y un rol de IAM personalizados que concedan permiso a los componentes de Image Builder para instalar paquetes de Systems Manager Distributor desde la instancia de compilación.

Paso 1: cree una política

Cree una política de IAM para los permisos de Distributor.

1. Abra la consola de IAM en https://console.aws.amazon.com/iam/.
2. En el panel de navegación, seleccione Políticas y, a continuación, Crear política.
3. En la página Crear política, elija la pestaña JSON y, a continuación, sustituya el contenido predeterminado por la siguiente política de JSON, sustituyendo la partición, la región y el identificador de cuenta según sea necesario, o bien utilizando caracteres comodín.

   JSON

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

```
    "Statement": [
        {
            "Sid": "AllowDistributorSendCommand",
            "Effect": "Allow",
            "Action": "ssm:SendCommand",
            "Resource": [
                "arn:aws:ssm:*::document/AWS-ConfigureAWSPackage",
                "arn:aws:ec2:*:111122223333:instance/*"
            ]
        },
        {
            "Sid": "AllowGetCommandInvocation",
            "Effect": "Allow",
            "Action": "ssm:GetCommandInvocation",
            "Resource": "*"
        }
    ]
}
```

4. Elija Review policy.
5. En Name (Nombre), escriba un nombre para identificar la política, por ejemplo, *InvokeDistributor* u otro nombre que prefiera.
6. (Opcional) En Description (Descripción), introduzca una descripción de la finalidad del rol.
7. Elija Create Policy.

Paso 2: cree un rol

Cree un rol de IAM para los permisos de Distributor.

1. En el panel de navegación de la consola de IAM, elija Roles y, a continuación, elija Crear rol.
2. En Select type of trusted entity (Seleccionar el tipo de entidad de confianza), elija Servicio de AWS.
3. Justo en Elija el servicio que utilizará esta función, elija y EC2, a continuación, elija Siguiente: permisos.
4. En Seleccione su caso de uso, elija y EC2, a continuación, elija Siguiente: permisos.
5. En la lista de políticas, selecciona la casilla de verificación situada junto a Amazon SSMManaged InstanceCore. (Escriba SSM en el cuadro de búsqueda si necesita reducir la lista.)

6. En esta lista de políticas, selecciona la casilla situada junto a EC2InstanceProfileForImageBuilder. (Escriba `ImageBuilder` en el cuadro de búsqueda si necesita reducir la lista.)
7. Elija Next: Tags (Siguiente: Etiquetas).
8. (Opcional) Agregue uno o varios pares clave-valor de etiqueta para organizar, realizar un seguimiento o controlar el acceso a este rol y, a continuación, elija Siguiente: revisar.
9. En Role name (Nombre del rol), escriba un nombre para el rol, por ejemplo, *`InvokeDistributor`* u otro nombre que prefiera.
10. (Opcional) En Role description (Descripción de rol), sustituya el texto predeterminado por una descripción de este rol.
11. Elija Create role. El sistema le devuelve a la página Roles.

Paso 3: asocie la política al rol

El último paso para configurar los permisos de Distributor es asociar la política de IAM al rol de IAM.

1. En la página Roles de la consola de IAM, seleccione el rol que acaba de crear. Se abre la página Summary (Resumen) del rol.
2. Seleccione Attach policies (Asociar políticas).
3. Busque la política creada en el procedimiento anterior y seleccione la casilla junto al nombre.
4. Elija Asociar política.

Utilice este rol en el recurso de configuración de infraestructura de Generador de Imágenes para cualquier imagen que incluya componentes que utilicen de Systems Manager Distributor. Para obtener más información, consulte Crear una configuración de infraestructura.

Configuración de **`distributor-package-windows`** como componente independiente

Para usar el componente `distributor-package-windows` en una receta, defina los siguientes parámetros que configuran el paquete que se va a instalar.

> **Note**
>
> Antes de utilizar el componente `distributor-package-windows` en una receta, debe asegurarse de que se cumplan todos los Requisitos previos.

- Acción (obligatorio): especifique si desea instalar o desinstalar el paquete. Los valores válidos son `Install` y `Uninstall`. Este valor se establece de forma predeterminada en `Install`.
- PackageName(Obligatorio): el nombre del paquete del distribuidor que se va a instalar o desinstalar. Para obtener una lista de nombres de paquete válidos, consulte Búsqueda de paquetes Distributor.
- PackageVersion(Opcional): la versión del paquete distribuidor que se va a instalar. PackageVersion toma de forma predeterminada la versión recomendada.
- AdditionalArguments(Opcional): cadena JSON que contiene los parámetros adicionales que debe proporcionar al script para instalar, desinstalar o actualizar un paquete. Para obtener más información, consulte additionalArguments en la sección Inputs (Entradas) de aws:configurePackage de la página de referencia del complemento del documento Systems Manager Command.

Configuración de **`aws-vss-components-windows`** como componente independiente

Al utilizar el componente `aws-vss-components-windows` en una receta, tiene la opción de configurar el parámetro `PackageVersion` para utilizar una versión específica del paquete `AwsVssComponents`. Si omite este parámetro, el componente utilizará de forma predeterminada la versión recomendada del paquete `AwsVssComponents`.

> **Note**
>
> Antes de utilizar el componente `aws-vss-components-windows` en una receta, debe asegurarse de que se cumplan todos los Requisitos previos.

Búsqueda de paquetes Distributor

Amazon y terceros proporcionan paquetes públicos que puede instalar con Systems Manager Distributor.

Para ver los paquetes disponibles en Consola de administración de AWS, inicie sesión en la AWS Systems Manager consola y seleccione Distribuidor en el panel de navegación. La página Distributor muestra todos los paquetes disponibles para usted. Para obtener más información sobre la lista de los paquetes disponibles en el AWS CLI, consulte Ver paquetes (línea de comandos) en la Guía del AWS Systems Manager usuario.

También puede crear sus propios paquetes de Systems Manager Distributor privados. Para obtener más información, consulte Crear un paquete en la Guía del usuario de AWS Systems Manager.

Componentes de endurecimiento de CIS

El Center for Internet Security (CIS, Centro para la seguridad de Internet) es una organización sin fines de lucro impulsada por la comunidad. Sus expertos en ciberseguridad trabajan juntos para desarrollar pautas de seguridad de TI que protejan a las organizaciones públicas y privadas contra las ciberamenazas. Su conjunto de mejores prácticas, reconocido a nivel mundial y conocido como Referencias del CIS, ayuda a las organizaciones de TI de todo el mundo a configurar sus sistemas de forma segura. Para ver artículos de actualidad, entradas de blog, podcasts, seminarios web y documentos técnicos, consulte CIS Insights en el sitio web del Center for Internet Security (Centro para la seguridad de Internet).

Referencias del CIS

CIS crea y mantiene un conjunto de pautas de configuración, conocidas como Referencias del CIS, que proporcionan las mejores prácticas de configuración para tecnologías específicas, incluidos los sistemas operativos, las plataformas en la nube, las aplicaciones, las bases de datos y más. Las Referencias del CIS se reconocen como un estándar de la industria por las organizaciones y estándares como PCI DSS, HIPAA, DoD Cloud Computing SRG, FISMA, DFARS y FEDRAMP. Para obtener más información, consulte Referencias del CIS en el sitio web del Center for Internet Security.

Componentes de endurecimiento de CIS

Cuando se suscribe a una imagen reforzada de CIS AWS Marketplace, también obtiene acceso al componente de refuerzo asociado que ejecuta un script para aplicar las directrices de nivel 1 de CIS Benchmarks para su configuración. La organización del CIS es propietaria y mantiene los componentes de endurecimiento del CIS para garantizar que reflejen las directrices más recientes.

> **Note**
>
> Los componentes de endurecimiento del CIS no siguen las reglas de ordenación de componentes estándar de las recetas de Generador de Imágenes. Los componentes de endurecimiento del CIS siempre se ejecutan en último lugar para garantizar que las pruebas de referencia se ejecuten con la imagen de salida.

Componentes de endurecimiento de STIG administrados por Amazon para Generador de imágenes

Las guías de implementación técnica de seguridad (STIGs) son los estándares que refuerzan la configuración creados por la Agencia de Sistemas de Información de Defensa (DISA) para proteger los sistemas de información y el software. Para hacer nuestros sistemas de conformidad con los estándares de STIG, debe instalar, configurar y probar varias configuraciones de seguridad.

Generador de Imágenes proporciona componentes de endurecimiento de STIG para crear de manera más eficiente imágenes compatibles con los estándares de referencia de STIG. Estos componentes de STIG escanean en busca de configuraciones incorrectas y ejecutan un script de corrección. El uso de componentes que cumplen con STIG no conlleva cargos adicionales.

> **Important**
>
> Con pocas excepciones, los componentes de endurecimiento de STIG no instalan paquetes de terceros. Si ya hay paquetes de terceros instalados en la instancia y si hay paquetes relacionados STIGs que Image Builder admite para ese paquete, el componente de refuerzo los aplica.

En esta página se enumeran todos los STIGs elementos compatibles con Image Builder que se aplican a las EC2 instancias que Image Builder lanza al crear y probar una imagen nueva. Si desea aplicar ajustes STIG adicionales a la imagen, puede crear un componente personalizado para configurarla. Para obtener más información sobre los componentes personalizados y cómo crearlos, consulte Uso de componentes para personalizar su imagen de Generador de imágenes.

Al crear una imagen, se aplica o se omite el registro de componentes de refuerzo de STIG, independientemente de si STIGs son compatibles. Le recomendamos que revise los registros de Generador de Imágenes para ver las imágenes que utilizan componentes de endurecimiento de

STIG. Para obtener más información sobre cómo acceder y revisar los registros de Generador de Imágenes, consulte Solución de problemas de canalizaciones.

Niveles de conformidad

- Alto (Categoría I)

  El riesgo más grave. Incluye cualquier vulnerabilidad que pueda resultar en la pérdida de confidencialidad, disponibilidad o integridad.
- Medio (Categoría II)

  Incluye cualquier vulnerabilidad que pueda resultar en la pérdida de confidencialidad, disponibilidad o integridad, pero los riesgos se pueden mitigar.
- Bajo (Categoría III)

  Cualquier vulnerabilidad que degrade las medidas de protección contra la pérdida de confidencialidad, disponibilidad o integridad.

Temas

- Componentes de endurecimiento de STIG de Windows
- Registro del historial de versiones de STIG para Windows
- Componentes de endurecimiento de STIG de Linux
- Registro del historial de versiones de STIG para Linux
- Componente validador de cumplimiento del SCAP

Componentes de endurecimiento de STIG de Windows

TOE de AWS Los componentes de refuerzo STIG de Windows están diseñados para servidores independientes y aplican una política de grupo local. Los componentes de refuerzo compatibles con STIG se instalan InstallRoot desde el Departamento de Defensa (DoD) en la infraestructura de Windows para descargar, instalar y actualizar los certificados del DoD. También eliminan los certificados innecesarios para mantener la conformidad con el STIG. Actualmente, las bases de referencia de STIG son compatibles con las siguientes versiones de Windows Server: 2012 R2, 2016, 2019 y 2022.

En esta sección se muestra la configuración actual de cada uno de los componentes de endurecimiento de STIG de Windows, seguida de un registro del historial de versiones.

STIG Low de Windows (Categoría III)

La siguiente lista contiene la configuración de STIG que el componente de endurecimiento aplica a la infraestructura. Si una configuración compatible no es aplicable a la infraestructura, el componente de endurecimiento omite esa configuración y continúa. Por ejemplo, es posible que algunas configuraciones de STIG no se apliquen a los servidores independientes. Las políticas específicas de la organización también pueden afectar la configuración aplicada por el componente de endurecimiento como, por ejemplo, pedir requisitos a los administradores para revisar la configuración de los documentos.

Para obtener una lista completa de Windows STIGs, consulte la biblioteca de documentos. STIGs Para obtener información acerca de cómo ver la lista completa, consulte Herramientas de visualización de STIG.

- Windows Server 2022 STIG versión 2, versión 4

  V-254335, V-254336, V-254337, V-254338, V-254351, V-254357, V-254363 y V-254481
- Windows Server 2019 STIG versión 3 versión 4

  V-205691, V-205819, V-205858, V-205859, V-205860, V-205870, V-205871 y V-205923.
- Windows Server 2016 STIG versión 2, versión 10

  V-224916, V-224917, V-224918, V-224919, V-224931, V-224942 y V-225060.
- Windows Server 2012 R2 MS STIG Version 3 Release 5

  V-225250, V-225318, V-225319, V-225324, V-225327, V-225328, V-225330, V-225331, V-225332, V-225333, V-225334, V-225335, V-225336, V-225342, V-225343, V-225355, V-225357, V-225358, V-225359, V-225360, V-225362, V-225363, V-225376, V-225394, V-225412, V-225459, V-225460, V-225462, V-225468, V-225473, V-225476, V-225479, V-225480, V-225481, V-225482, V-225483, V-225484, V-225485, V-225487, V-225488, V-225489, V-225490, V-225511, V-225514, V-225525, V-225526, V-225536 y V-225537
- Microsoft.NET Framework 4.0 STIG versión 2 versión 6

  No se aplica ninguna configuración de STIG a Microsoft .NET Framework para vulnerabilidades de categoría III.
- Windows Firewall STIG, versión 2, versión 2

  V-241994, V-241995, V-241996, V-241999, V-242000, V-242001, V-242006, V-242007 y V-242008
- Internet Explorer 11 STIG versión 2 versión 5

V-223016, V-223056 y V-223078

- Microsoft Edge STIG versión 2 (solo Windows Server 2022)

  V-235727, V-235731, V-235751, V-235752 y V-235765

STIG Medium de Windows (Categoría II)

La siguiente lista contiene la configuración de STIG que el componente de endurecimiento aplica a la infraestructura. Si una configuración compatible no es aplicable a la infraestructura, el componente de endurecimiento omite esa configuración y continúa. Por ejemplo, es posible que algunas configuraciones de STIG no se apliquen a los servidores independientes. Las políticas específicas de la organización también pueden afectar la configuración aplicada por el componente de endurecimiento como, por ejemplo, pedir requisitos a los administradores para revisar la configuración de los documentos.

Para obtener una lista completa de Windows STIGs, consulte la biblioteca de STIGs documentos. Para obtener información acerca de cómo ver la lista completa, consulte Herramientas de visualización de STIG.

> **Note**
>
> Los componentes de protección STIG Medium de Windows incluyen todas las configuraciones STIG enumeradas que TOE de AWS se aplican a los componentes STIG de baja resistencia de Windows, además de las configuraciones STIG que se enumeran específicamente para las vulnerabilidades de categoría II.

- STIG de Windows Server 2022, versión 2, versión 4

  Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de categoría III (baja), además de:

  V-254247, V-254265, V-254269, V-254270, V-254271, V-254272, V-254273, V-254274, V-254276, V-254277, V-254278, V-254285, V-254286, V-254287, V-254288, V-254289, V-254290, V-254291, V-254292 254300, V-254301, V-254302, V-254303, V-254304, V-254305, V-254306, V-254307, V-254308, V-254309, V-254310, V-254311, V-254312, V-254313, V-254314, V-254315, V-254316, V-254317, V-254318, V-254318 254319, V-254320, V-254321, V-254322, V-254323, V-254324, V-254325, V-254326, V-254327, V-254328, V-254329, V-254330, V-254331, V-254332, V-254333,

V-254334, V-254339, V-254342 V-254344, V-254345, V-254346, V-254347, V-254348, V-254349, V-254350, V-254355, V-254356, V-254356, V-254358, V-254359, V-254360, V-254361, V-254362, V-254364, V-254365, V-254366, V-254367, V-254367 4368, V-254369, V-254370, V-254372, V-254373, V-254375, V-254376, V-254377, V-254379, V-254380, V-254382, V-254383, V-254384, V-254431, V-254432, V-254433, V-254434, V-254435, V-254435 36, V-254438, V-254439, V-254442, V-254443, V-254444, V-254445, V-254449, V-254450, V-254451, V-254452, V-254453, V-254454, V-254455, V-254456, V-254459, V-254460, V-254461, V-254462, V-254463, V-254464, V-254468, V-254470, V-254471, V-254472, V-254473, V-254476, V-254477, V-254478, V-254479, V-254480, V-254482, V-254483, V-254484, V-254485, V-254486, V-254487, V-254488, V-254489, V-254490, V-254493, V-254494, V-254495, V-254497, V-254499, V-254501, V-254502, V-254503, V-254504, V-254505, V-254507, V-254508, V-254509, V-254510, V-254511 y V-254512

- Windows Server 2019 STIG versión 3 versión 4

  Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de categoría III (baja), además de:

  V-205625, V-205626, V-205627, V-205629, V-205630, V-205633, V-205634, V-205635, V-205636, V-205637, V-205638, V-205639, V-205643, V-205644, V-205648, V-205649, V-205650, V-205651, V-205652, V-205652 205655, V-205656, V-205659, V-205662, V-205671, V-205672, V-205673, V-205675, V-205676, V-205678, V-205679, V-205680, V-205681, V-205682, V-205683, V-205684, V-205685, V-205686, V-205686 205687, V-205688, V-205689, V-205690, V-205692, V-205693, V-205694, V-205697, V-205698, V-205708, V-205709, V-205712, V-205714, V-205716, V-205717, V-205718, V-205719, V-205720, V-205722, V-205729, V-205730, V-205733, V-205747, V-205751, V-205752, V-205754, V-205756, V-205758, V-205759, V-205760, V-205761, V-205762, V-205764, V-205765, V-205766, V-205767, V-205768, V-205769, V-205770, V-205771, V-205772, V-205773, V-205774, V-205775, V-205776, V-205777, V-205778, V-205779, V-205780, V-205781, V-205782, V-205783, V-205784, V-205795, V-205796, V-205797, V-205798, V-205798 205801, V-205808, V-205809, V-205810, V-205811, V-205812, V-205813, V-205814, V-205815, V-205816, V-205817, V-205821, V-205822, V-205823, V-205824, V-205825, V-205826, V-205827, V-205828, V-205830, V-205832, V-205833, V-205834, V-205835, V-205836, V-205837, V-205838, V-205839, V-205840, V-205841, V-205842, V-205861, V-205863, V-205865, V-205866, V-205867, V-205868, V-205869, V-205872, V-205873, V-205874, V-205911, V-205912, V-205915, V-205916, V-205917, V-205918, V-205920, V-205921, V-205922, V-205924, V-205925, V-236001 y V-257503

- Windows Server 2016 STIG versión 2 versión 10

  Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de categoría III (baja), además de:

V-224850, V-224852, V-224853, V-224854, V-224855, V-224856, V-224857, V-224858, V-224859, V-224866, V-224867, V-224868, V-224869, V-224870, V-224871, V-224872, V-224873, V-224881, V-224882, V-224883, V-224884, V-224885, V-224887, V-224888, V-224889, V-224890, V-224891, V-224892, V-224893, V-224894, V-224895, V-224896, V-224897, V-224898, V-224899, V-224900, V-224901, V-224901 224902, V-224903, V-224904, V-224905, V-224906, V-224907, V-224908, V-224909, V-224910, V-224911, V-224912, V-224913, V-224914, V-224915, V-224920, V-224922, V-224924, V-224925, V-224926, V-224927, V-224928, V-224929, V-224935, V-224936, V-224937, V-224938, V-224939, V-224940, V-224941, V-224943, V-224944, V-224945, V-224946, V-224947, V-224948, V-224949, V-224951 V-224952, V-224953, V-224955, V-224956, V-224957, V-224959, V-224960, V-224962, V-224963, V-225010, V-225013, V-225014, V-225015, V-225016, V-225017, V-225018, V-225019, V-225021, V-225022 V-225023, V-225024, V-225028, V-225029, V-225030, V-225031, V-225032, V-225033, V-225034, V-225035, V-225038, V-225039, V-225040, V-225041, V-225042, V-225043, V-225047, V-225049, V-2250, V-2250, V-225051, V-225052, V-225055, V-225056, V-225057, V-225058, V-225059, V-225061, V-225062, V-225063, V-225064, V-225065, V-225066, V-225067, V-225068, V-225069, V-225072, V-225073, V-225074, V-225076, V-225078, V-225080, V-225081, V-225082, V-225083, V-225084, V-225086, V-225087, V-225088, V-225089, V-225092, V-225093, V-236000 y V-257502

- Windows Server 2012 R2 MS STIG versión 3, versión 5

  Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de categoría III (baja), además de:

  V-225239, V-225259, V-225260, V-225261, V-225263, V-225264, V-225265, V-225266, V-225267, V-225269, V-225270, V-225271, V-225272, V-225273, V-225275, V-225276, V-225277, V-225278, V-225279, V-225280, V-225281, V-225282, V-225284, V-225285, V-225286, V-225287, V-225288, V-225289, V-225290, V-225291, V-225292, V-225293, V-225294, V-225295, V-225296, V-225297, V-225298, V-225298 225299, V-225300, V-225301, V-225303, V-225304, V-225305, V-225314, V-225315, V-225316, V-225317, V-225325, V-225326, V-225329, V-225337, V-225338, V-225339, V-225340, V-225341, V-225344, V-225345, V-225347, V-225348, V-225349, V-225350, V-225351, V-225352, V-225353, V-225356, V-225367, V-225368, V-225369, V-225370, V-225371, V-225372, V-225373, V-225374, V-225375, V-225378, V-225379, V-225380, V-225381, V-225382, V-225383, V-225384, V-225385, V-225386, V-225389, V-225391, V-225393, V-225395, V-225397, V-225398, V-225400, V-225400, V-225401, V-225402, V-225404, V-225406, V-225407, V-225408, V-225409, V-225410, V-225411, V-225413, V-225414, V-225415, V-225441, V-225442, V-225443, V-225448, V-225452, V-225453 V-225454, V-225455, V-225456, V-225457, V-225458, V-225461, V-225463, V-225464, V-225469, V-225470, V-225471, V-225472, V-225474, V-225475, V-225477, V-225478,

V-225486, V-225494, V-225500, V-225500 225501, V-225502, V-225503, V-225504, V-225506, V-225508, V-225509, V-225510, V-225513, V-225515, V-225516, V-225517, V-225518, V-225519, V-225520, V-225521, V-225522, V-225523, V-225524, V-225524 225527, V-225528, V-225529, V-225530, V-225531, V-225532, V-225533, V-225534, V-225535, V-225538, V-225539, V-225540, V-225541, V-225542, V-225543, V-225544, V-225545, V-225546, V-225548, V-225549, V-225550, V-225551, V-225553, V-225554, V-225555, V-225557, V-225558, V-225559, V-225560, V-225561, V-225562, V-225563, V-225564, V-225565, V-225566, V-225567, V-225568, V-225569, V-225569 225570, V-225571, V-225572, V-225573 y V-225574

- Microsoft.NET Framework 4.0 STIG versión 2 versión 6

  Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de categoría III (baja), además de V-225238.

- Windows Firewall STIG, versión 2, versión 2

  Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de categoría III (baja), además de:

  V-241989, V-241990, V-241991, V-241993, V-241993, V-241998 y V-242003

- Internet Explorer 11 STIG versión 2 versión 5

  Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de categoría III (baja), además de:

  V-223015, V-223017, V-223018, V-223019, V-223020, V-223021, V-223022, V-223023, V-223024, V-223025, V-223026, V-223027, V-223028, V-223029, V-223030, V-223031, V-223032, V-223033, V-223034, V-223035, V-223036, V-223037, V-223038, V-223039, V-223040, V-223041, V-223042, V-223043, V-223044, V-223045, V-223046, V-223048, V-223049, V-223050, V-223051, V-223052, V-223053, V-223054, V-223055, V-223057, V-223058, V-223059, V-223060, V-223061, V-223062, V-223063, V-223064, V-223065, V-223066, V-223067, V-223068, V-223069, V-223070, V-223071, V-223072, V-223073, V-223074, V-223075, V-223076, V-223077, V-223080, V-223081, V-223082, V-223083, V-223084, V-223085, V-223086, V-223087, V-223088, V-223089, V-223090, V-223091, V-223092, V-223093, V-223094, V-223095, V-223096, V-223097, V-223098, V-223099, V-223100, V-223101, V-223102, V-223103, V-223104, V-223105, V-223106, V-223107, V-223108, V-223109, V-223110, V-223111, V-223112, V-223113, V-223114, V-223115, V-223116, V-223117, V-223118, V-223119, V-223120, V-223121, V-223122, V-223123, V-223124, V-223125, V-223126, V-223127, V-223128, V-223129, V-223130, V-223131, V-223132, V-223133, V-223134, V-223135, V-223136, V-223137, V-223138, V-223139, V-223140, V-223141, V-223142, V-223143, V-223144, V-223145, V-223146, V-223147, V-223148, V-223149, V-250540 y V-250541

- Microsoft Edge STIG versión 2 (solo Windows Server 2022)

  Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de categoría III (baja), además de:

  V-235720, V-235721, V-235723, V-235724, V-235725, V-235726, V-235728, V-235729, V-235730, V-235732, V-235733, V-235734, V-235735, V-235736, V-235737, V-235738, V-235739, V-235740, V-235741, V-235742, V-235743, V-235744, V-235745, V-235746, V-235747, V-235748, V-235749, V-235750, V-235754, V-235756, V-235760, V-235761, V-235763, V-235764, V-235766, V-235767, V-235768, V-235769, V-235770, V-235771, V-235772, V-235773, V-235774 y V-246736

- Microsoft Defender STIG versión 2 versión 4

  Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de categoría III (baja), además de:

  V-213427, V-213429, V-213430, V-213431, V-213432, V-213433, V-213434, V-213435, V-213436, V-213437, V-213438, V-213439, V-213440, V-213441, V-213442, V-213443, V-213444, V-213445, V-213446, V-213447, V-213448, V-213449, V-213450, V-213451, V-213455, V-213464, V-213465 y V-213466

STIG de Windows versión I (Categoría I)

La siguiente lista contiene la configuración de STIG que el componente de endurecimiento aplica a la infraestructura. Si una configuración compatible no es aplicable a la infraestructura, el componente de endurecimiento omite esa configuración y continúa. Por ejemplo, es posible que algunas configuraciones de STIG no se apliquen a los servidores independientes. Las políticas específicas de la organización también pueden afectar la configuración aplicada por el componente de endurecimiento como, por ejemplo, pedir requisitos a los administradores para revisar la configuración de los documentos.

Para obtener una lista completa de Windows STIGs, consulte la biblioteca de STIGs documentos. Para obtener información acerca de cómo ver la lista completa, consulte Herramientas de visualización de STIG.

> **Note**
>
> Los componentes de protección STIG High de Windows incluyen todas las configuraciones de STIG enumeradas que TOE de AWS se aplican a los componentes de protección de

Windows STIG Low y Windows STIG Medium, además de las configuraciones de STIG que se enumeran específicamente para las vulnerabilidades de categoría I.

- Windows Server 2022 (STIG, versión 2, versión 4)

  Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de las categorías II y III (media y baja), además de:

  V-254293, V-254352, V-254353, V-254354, V-254374, V-254378, V-254381, V-254446, V-254465, V-254466, V-254467, V-254469, V-254474, V-254475 y V-254500
- Windows Server 2019 STIG versión 3 versión 4

  Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de las categorías II y III (media y baja), además de:

  V-205653, V-205654, V-205711, V-205713, V-205724, V-205725, V-205757, V-205802, V-205804, V-205805, V-205806, V-205849, V-205908, V-205913, V-205914 y V-205919
- Windows Server 2016 STIG versión 2, versión 10

  Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de las categorías II y III (media y baja), además de:

  V-224874, V-224932, V-224933, V-224934, V-224954, V-224958, V-224961, V-225025, V-225044, V-225045, V-225046, V-225048, V-225053, V-225054 y V-225079
- Windows Server 2012 R2 MS STIG versión 3, versión 5

  Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de las categorías II y III (media y baja), además de:

  V-225274, V-225354, V-225365, V-225366, V-225390, V-225396, V-225399, V-225444, V-225449, V-225491, V-225492, V-225493, V-225496, V-225497, V-225498, V-225505, V-225507, V-225547, V-225547 5552 y V-225556
- Microsoft.NET Framework 4.0 STIG versión 2 versión 6

  Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de las categorías II y III (media y baja) para Microsoft.NET Framework. No se aplica ninguna configuración de STIG adicional para vulnerabilidades de categoría I.

- Windows Firewall STIG, versión 2, versión 2

  Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de las categorías II y III (media y baja), además de:

  V-241992, V-241997 y V-242002
- Internet Explorer 11 STIG versión 2 versión 5

  V-252910
- Microsoft Edge STIG versión 2 (solo Windows Server 2022)

  Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de las categorías II y III (media y baja), además de:

  V-235758 y V-235759
- Microsoft Defender STIG versión 2 versión 4

  Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de las categorías II y III (media y baja), además de:

  V-213426, V-213452 y V-213453

Registro del historial de versiones de STIG para Windows

Esta sección registra el historial de versiones de los componentes de endurecimiento de Windows para las actualizaciones trimestrales de STIG. Para ver los cambios y las versiones publicadas durante un trimestre, elija el título para ampliar la información.

Cambios en el tercer trimestre de 2025:9 de abril de 2025 (sin cambios):

No hubo cambios en el STIGS, componente de Windows, para la versión del tercer trimestre de 2025.

Cambios en el segundo trimestre de 2025:26 de junio de 2025:

Se actualizaron las versiones de STIG y se aplicó el STIGS para la versión del segundo trimestre de 2025 de la siguiente manera:

STIG-Build-Windows-Low versión 2025.2.x

- Windows Server 2022 STIG versión 2 versión 4

- Windows Server 2019 STIG versión 3 versión 4
- Windows Server 2016 STIG versión 2, versión 10
- Windows Server 2012 R2 MS STIG Version 3 Release 5
- Microsoft.NET Framework 4.0 STIG versión 2 versión 6
- Windows Firewall STIG, versión 2, versión 2
- Internet Explorer 11 STIG versión 2 versión 5
- Microsoft Edge STIG versión 2 (solo Windows Server 2022)

STIG-Build-Windows-Medium versión 2025.2.x

- Windows Server 2022 STIG versión 2 versión 4
- Windows Server 2019 STIG versión 3 versión 4
- Windows Server 2016 STIG versión 2, versión 10
- Windows Server 2012 R2 MS STIG Version 3 Release 5
- Microsoft.NET Framework 4.0 STIG versión 2 versión 6
- Windows Firewall STIG, versión 2, versión 2
- Internet Explorer 11 STIG versión 2 versión 5
- Microsoft Edge STIG versión 2 (solo Windows Server 2022)
- Defender STIG versión 2, versión 4

STIG-Build-Windows-High versión 2025.2.x

- Windows Server 2022 STIG versión 2 versión 4
- Windows Server 2019 STIG versión 3 versión 4
- Windows Server 2016 STIG versión 2, versión 10
- Windows Server 2012 R2 MS STIG Version 3 Release 5
- Microsoft.NET Framework 4.0 STIG versión 2 versión 6
- Windows Firewall STIG, versión 2, versión 2
- Internet Explorer 11 STIG versión 2 versión 5
- Microsoft Edge STIG versión 2 (solo Windows Server 2022)
- Defender STIG versión 2, versión 4

Cambios en el primer trimestre de 2025:5 de abril de 2025:

Se actualizó STIGS para Internet Explorer 11, STIG, versión 2, versión 5, para todos los componentes de STIG para la versión del primer trimestre de 2025.

- STIG-Build-Windows-Low versión 2025.1.x
- STIG-Build-Windows-Medium versión 2025.1.x
- STIG-Build-Windows-High versión 2025.1.x

Cambios en el cuarto trimestre de 2024:2 de abril de 2025:

Se actualizaron las versiones de STIG y se aplicó el STIGS para la versión del cuarto trimestre de 2024 de la siguiente manera:

STIG-Build-Windows-Low versión 2024.4.0

- Windows Server 2022 STIG versión 2 versión 2
- Windows Server 2019 STIG versión 3, versión 2
- Windows Server 2016 STIG versión 2 versión 9
- Windows Server 2012 R2 MS STIG versión 3, versión 5
- Microsoft .NET Framework 4.0 STIG versión 2, versión 2
- Windows Firewall STIG, versión 2, versión 2
- Internet Explorer 11 STIG versión 2 versión 5
- Microsoft Edge STIG versión 2 (solo Windows Server 2022)

STIG-Build-Windows-Medium versión 2024.4.0

- Windows Server 2022 STIG versión 2 versión 2
- Windows Server 2019 STIG versión 3, versión 2
- Windows Server 2016 STIG versión 2 versión 9
- Windows Server 2012 R2 MS STIG versión 3, versión 5
- Microsoft .NET Framework 4.0 STIG versión 2, versión 2
- Windows Firewall STIG, versión 2, versión 2
- Internet Explorer 11 STIG versión 2 versión 5
- Microsoft Edge STIG versión 2 (solo Windows Server 2022)

- Defender STIG versión 2, versión 4

STIG-Build-Windows-High versión 2024.4.0

- Windows Server 2022 STIG versión 2 versión 2
- Windows Server 2019 STIG versión 3, versión 2
- Windows Server 2016 STIG versión 2 versión 9
- Windows Server 2012 R2 MS STIG versión 3, versión 5
- Microsoft .NET Framework 4.0 STIG versión 2, versión 2
- Windows Firewall STIG, versión 2, versión 2
- Internet Explorer 11 STIG versión 2 versión 5
- Microsoft Edge STIG versión 2 (solo Windows Server 2022)
- Defender STIG versión 2, versión 4

Cambios en el tercer trimestre de 2024:4 de octubre de 2023 (sin cambios):

No hubo cambios en el STIGS, componente de Windows, en la versión del tercer trimestre de 2024.

Cambios en el segundo trimestre de 2024: 10 de mayo de 2024 (sin cambios):

No se produjeron cambios en el componente STIGS de Windows para la versión del segundo trimestre de 2024.

Cambios en el primer trimestre de 2024: 6 de febrero de 2024 (sin cambios):

No se produjeron cambios en el componente STIGS de Windows para la versión del primer trimestre de 2024.

Cambios en el cuarto trimestre de 2023: 4 de diciembre de 2023 (sin cambios):

No se produjeron cambios en el componente STIGS de Windows para la versión del cuarto trimestre de 2023.

Cambios en el tercer trimestre de 2023: 4 de octubre de 2023 (sin cambios):

No se produjeron cambios en el componente STIGS de Windows para la versión del tercer trimestre de 2023.

Cambios en el segundo trimestre de 2023: 3 de mayo de 2023 (sin cambios):

No se produjeron cambios en el componente STIGS de Windows para la versión del segundo trimestre de 2023.

Cambios en el primer trimestre de 2023: 27 de marzo de 2023 (sin cambios):

No se produjeron cambios en el componente STIGS de Windows para la versión del primer trimestre de 2023.

Cambios en el cuarto trimestre de 2022: 1 de febrero de 2023:

Se actualizaron las versiones de STIG y se aplicó el STIGS para la versión del cuarto trimestre de 2022 de la siguiente manera:

STIG-Build-Windows-Low versión 2022.4.x

- Windows Server 2022 STIG versión 1, versión 1
- Windows Server 2019 STIG versión 2, versión 5
- Windows Server 2016 STIG versión 2, versión 5
- Windows Server 2012 R2 MS STIG versión 3, versión 5
- Microsoft .NET Framework 4.0 STIG versión 2, versión 2
- Firewall de Windows STIG versión 2, versión 1
- Internet Explorer 11 STIG versión 2, versión 3
- STIG de Microsoft Edge versión 1, lanzamiento 6 (solo Windows Server 2022)

STIG-Build-Windows-Medium versión 2022.4.x

- Windows Server 2022 STIG versión 1, versión 1
- Windows Server 2019 STIG versión 2, versión 5
- Windows Server 2016 STIG versión 2, versión 5
- Windows Server 2012 R2 MS STIG versión 3, versión 5
- Microsoft .NET Framework 4.0 STIG versión 2, versión 2
- Firewall de Windows STIG versión 2, versión 1
- Internet Explorer 11 STIG versión 2, versión 3
- STIG de Microsoft Edge versión 1, lanzamiento 6 (solo Windows Server 2022)
- STIG de Defender versión 2, lanzamiento 4 (solo Windows Server 2022)

STIG-Build-Windows-High versión 2022.4.x

- Windows Server 2022 STIG versión 1, versión 1
- Windows Server 2019 STIG versión 2, versión 5
- Windows Server 2016 STIG versión 2, versión 5
- Windows Server 2012 R2 MS STIG versión 3, versión 5
- Microsoft .NET Framework 4.0 STIG versión 2, versión 2
- Firewall de Windows STIG versión 2, versión 1
- Internet Explorer 11 STIG versión 2, versión 3
- STIG de Microsoft Edge versión 1, lanzamiento 6 (solo Windows Server 2022)
- STIG de Defender versión 2, lanzamiento 4 (solo Windows Server 2022)

Cambios en el tercer trimestre de 2022: 30 de septiembre de 2022 (sin cambios):

No se produjeron cambios en el componente STIGS de Windows para la versión del tercer trimestre de 2022.

Cambios en el segundo trimestre de 2022: 2 de agosto de 2022:

Se actualizaron las versiones de STIG y se aplicó el STIGS para la versión del segundo trimestre de 2022.

STIG-Build-Windows-Low versión 1.5.x

- Windows Server 2019 STIG versión 2, lanzamiento 4
- Windows Server 2016 STIG versión 2, lanzamiento 4
- Windows Server 2012 R2 MS STIG Version 3 Release 3
- Microsoft .NET Framework 4.0 STIG Version 2 Release 1
- Firewall de Windows STIG versión 2, versión 1
- Internet Explorer 11 STIG Version 1 Release 19

STIG-Build-Windows-Medium versión 1.5.x

- Windows Server 2019 STIG versión 2, lanzamiento 4
- Windows Server 2016 STIG versión 2, lanzamiento 4
- Windows Server 2012 R2 MS STIG Version 3 Release 3

- Microsoft .NET Framework 4.0 STIG Version 2 Release 1
- Firewall de Windows STIG versión 2, versión 1
- Internet Explorer 11 STIG Version 1 Release 19

STIG-Build-Windows-High versión 1.5.x

- Windows Server 2019 STIG versión 2, lanzamiento 4
- Windows Server 2016 STIG versión 2, lanzamiento 4
- Windows Server 2012 R2 MS STIG Version 3 Release 3
- Microsoft .NET Framework 4.0 STIG Version 2 Release 1
- Firewall de Windows STIG versión 2, versión 1
- Internet Explorer 11 STIG Version 1 Release 19

Cambios en el primer trimestre de 2022: 2 de agosto de 2022 (sin cambios):

No se produjeron cambios en el componente STIGS de Windows para la versión del primer trimestre de 2022.

Cambios en el cuarto trimestre de 2021: 20 de diciembre de 2021:

Se actualizaron las versiones de STIG y se aplicó el STIGS para la versión del cuarto trimestre de 2021.

STIG-Build-Windows-Low versión 1.5.x

- Windows Server 2019 STIG Version 2 Release 3
- Windows Server 2016 STIG Version 2 Release 3
- Windows Server 2012 R2 MS STIG Version 3 Release 3
- Microsoft .NET Framework 4.0 STIG Version 2 Release 1
- Firewall de Windows STIG versión 2, versión 1
- Internet Explorer 11 STIG Version 1 Release 19

STIG-Build-Windows-Medium versión 1.5.x

- Windows Server 2019 STIG Version 2 Release 3
- Windows Server 2016 STIG Version 2 Release 3

- Windows Server 2012 R2 MS STIG Version 3 Release 3
- Microsoft .NET Framework 4.0 STIG Version 2 Release 1
- Firewall de Windows STIG versión 2, versión 1
- Internet Explorer 11 STIG Version 1 Release 19

STIG-Build-Windows-High versión 1.5.x

- Windows Server 2019 STIG Version 2 Release 3
- Windows Server 2016 STIG Version 2 Release 3
- Windows Server 2012 R2 MS STIG Version 3 Release 3
- Microsoft .NET Framework 4.0 STIG Version 2 Release 1
- Firewall de Windows STIG versión 2, versión 1
- Internet Explorer 11 STIG Version 1 Release 19

Cambios en el tercer trimestre de 2021: 30 de septiembre de 2021:

Se actualizaron las versiones de STIG y se aplicó el STIGS para la versión del tercer trimestre de 2021.

STIG-Build-Windows-Low versión 1.4.x

- Windows Server 2019 STIG Version 2 Release 2
- Windows Server 2016 STIG Version 2 Release 2
- Windows Server 2012 R2 MS STIG Version 3 Release 2
- Microsoft .NET Framework 4.0 STIG Version 2 Release 1
- Windows Firewall STIG Version 1 Release 7
- Internet Explorer 11 STIG versión 1, versión 19

STIG-Build-Windows-Medium versión 1.4.x

- Windows Server 2019 STIG Version 2 Release 2
- Windows Server 2016 STIG Version 2 Release 2
- Windows Server 2012 R2 MS STIG Version 3 Release 2
- Microsoft .NET Framework 4.0 STIG Version 2 Release 1

- Windows Firewall STIG Version 1 Release 7
- Internet Explorer 11 STIG versión 1, versión 19

STIG-Build-Windows-High versión 1.4.x

- Windows Server 2019 STIG Version 2 Release 2
- Windows Server 2016 STIG Version 2 Release 2
- Windows Server 2012 R2 MS STIG Version 3 Release 2
- Microsoft .NET Framework 4.0 STIG Version 2 Release 1
- Windows Firewall STIG Version 1 Release 7
- Internet Explorer 11 STIG versión 1, versión 19

Componentes de endurecimiento de STIG de Linux

Esta sección contiene información sobre los componentes de endurecimiento de STIG de Linux, seguida de un registro del historial de versiones. Si la distribución de Linux no tiene una configuración de STIG propia, el componente de endurecimiento aplica la configuración de RHEL.

Los componentes de Linux tienen parámetros de entrada opcionales que le ayudan a personalizar los siguientes comportamientos para su instancia de Linux.

- InstallPackages (cadena) Si el valor es`No`, el componente no instala ningún paquete de software adicional. Si el valor es`Yes`, el componente instala los paquetes de software adicionales necesarios para garantizar la máxima conformidad. El valor predeterminado es `No`.
- SetDoDConsentBanner (cadena) Si el valor es`No`, el banner de consentimiento del DoD no se muestra cuando se conecta a una instancia que tiene instalado uno de los componentes de STIG Linux. Si el valor es`Yes`, se muestra el banner de consentimiento del DoD antes de iniciar sesión cuando se conecta a una instancia que tiene instalado uno de los componentes de STIG Linux. Debe reconocer el banner antes de iniciar sesión. El valor predeterminado es `No`.

  Para ver un ejemplo del banner de consentimiento, consulte el Aviso de exención de responsabilidad sobre privacidad y consentimiento del Departamento de Defensa que aparece cuando accede al sitio web de DLA Document Services.

El componente de endurecimiento aplica la configuración de STIG compatible a la infraestructura basada en la distribución de Linux, de la siguiente manera:

Configuración de STIG de Red Hat Enterprise Linux (RHEL) 7

- RHEL 7
- CentOS 7
- Amazon Linux (2AL2)

Configuración de RHEL 8 STIG

- RHEL 8
- CentOS 8

Configuración de RHEL 9 STIG

- RHEL 9
- CentOS Stream 9

STIG de Linux versión III (Categoría III)

La siguiente lista contiene la configuración de STIG que el componente de endurecimiento aplica a la infraestructura. Si una configuración compatible no es aplicable a la infraestructura, el componente de endurecimiento omite esa configuración y continúa. Por ejemplo, es posible que algunas configuraciones de STIG no se apliquen a los servidores independientes. Las políticas específicas de la organización también pueden afectar la configuración aplicada por el componente de endurecimiento como, por ejemplo, pedir requisitos a los administradores para revisar la configuración de los documentos.

Para obtener una lista completa, consulte la biblioteca de STIGs documentos. Para obtener información acerca de cómo ver la lista completa, consulte Herramientas de visualización de STIG.

RHEL 7 STIG, versión 3, versión 15

- RHEL 7/CentOS 7/ AL2

  V-204452, V-204576 y V-204605

RHEL 8 STIG versión 2 versión 4

- RHEL 8/CentOS 8

V-230241, V-230269, V-230270, V-230281, V-230285, V-230346, V-230381, V-230395, V-230468, V-230469, V-230485, V-230486, V-230491, V-230494, V-230495, V-230496, V-230497, V-230498, V-230499, y V-244527

RHEL 9 STIG versión 2 versión 5

- RHEL 9/CentOS Stream 9

  V-257782, V-257795, V-257796, V-257824, V-257880, V-257946, V-257947, V-258037, V-258067, V-258069, V-258076, V-258138 y V-258173

Amazon Linux 2023 STIG versión 1 versión 1

V-274141

SLES 12 STIG versión 3 versión 3

V-217108, V-217113, V-217140, V-217198, V-217209, V-217211, V-217212, V-217213, V-217214, V-217215, V-217216, V-217236, V-217237, V-217238, V-217239, V-217282 y V-255915

SLES 15 STIG versión 2, versión 5

V-234811, V-234850, V-234868, V-234873, V-234873, V-234905, V-234907, V-234908, V-234909, V-234933, V-234934, V-234935, V-234936, V-234955, V-234963, V-234967 y V-255921

Ubuntu 18.04 STIG versión 2, versión 15

V-219163, V-219164, V-219165, V-219172, V-219173, V-219174, V-219175, V-219178, V-219179, V-219180, V-219210, V-219301, V-219327, V-219332 y V-219333

Ubuntu 20.04 STIG versión 2, versión 3

V-238202, V-238203, V-238221, V-238222, V-238223, V-238224, V-238226, V-238234, V-238235, V-238237, V-238308, V-238323, V-238357, V-238362 y V-238373

Ubuntu 22.04 STIG versión 2, versión 5

V-260479, V-260480, V-260481, V-260521, V-260520, V-260476, V-260472, V-260549, V-260550, V-260551, V-260552, V-260581 y V-260596

Ubuntu 24.04 STIG versión 1, versión 2

V-270645, V-270646, V-270664, V-270820, V-270677, V-270690, V-270706, V-270710, V-270695, V-270749, V-270752 y V-270818

STIG de Linux versión II (Categoría II)

La siguiente lista contiene la configuración de STIG que el componente de endurecimiento aplica a la infraestructura. Si una configuración compatible no es aplicable a la infraestructura, el componente de endurecimiento omite esa configuración y continúa. Por ejemplo, es posible que algunas configuraciones de STIG no se apliquen a los servidores independientes. Las políticas específicas de la organización también pueden afectar la configuración aplicada por el componente de endurecimiento como, por ejemplo, pedir requisitos a los administradores para revisar la configuración de los documentos.

Para obtener una lista completa STIGs , consulte la biblioteca de documentos. Para obtener información acerca de cómo ver la lista completa, consulte Herramientas de visualización de STIG.

> **Note**
>
> Los componentes de protección STIG Medium de Linux incluyen todas las configuraciones de STIG que se indican en la lista y que TOE de AWS se aplican a los componentes de protección baja de STIG de Linux, además de las configuraciones de STIG que se enumeran específicamente para las vulnerabilidades de categoría II.

RHEL 7 STIG, versión 3, versión 15

Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de categoría III (baja) de esta distribución de Linux, además de:

- RHEL 7/CentOS 7/ AL2

  V-204405, V-204406, V-204407, V-204408, V-204409, V-204410, V-204411, V-204412, V-204413, V-204414, V-204415, V-204416, V-204417, V-204418, V-204420, V-204422, V-204423, V-204426, V-204427, V-204431, V-204434, V-204435, V-204437, V-204449, V-204450, V-204451, V-204457, V-204466, V-204490, V-204491, V-204503, V-204507, V-204508, V-204510, V-204511, V-204512, V-204514, V-204515 V-204516, V-204517, V-204521, V-204524, V-204531, V-204536, V-204537, V-204538, V-204539, V-204540, V-204541, V-204542, V-204543, V-204544, V-204545, V-204546, V-204547, V-204548, V-204549, V-204550, V-204551, V-204552, V-204553, V-204554, V-204555, V-204556, V-204557, V-204558, V-204559, V-204560, V-204562, V-204563, V-204564, V-204565,

V-204566, V-204567, V-204568, V-204572, V-2020V-204584, V-204585, V-204587, V-204588, V-204589, V-204590, V-204591, V-204592, V-204593, V-204596, V-204597, V-204598, V-204599, V-204600, V-204601, V-204602, V-204609, V-204610, V-202010 4611, V-204612, V-204613, V-204614, V-204615, V-204616, V-204617, V-204619, V-204622, V-204625, V-204630, V-204631, V-204633, V-233307, V-237634, V-237635, V-251703, V-255925, V-255927, y V-256970

RHEL 8 STIG, versión 2, versión 4

Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de categoría III (baja) de esta distribución de Linux, además de:

- RHEL 8/CentOS 8

V-230222, V-230228, V-230231, V-230233, V-230236, V-230237, V-230238, V-230239, V-230240, V-230244, V-230245, V-230246, V-230247, V-230248, V-230249, V-230250, V-230255, V-230256, V-230257 V-230258, V-230259, V-230260, V-230261, V-230262, V-230266, V-230267, V-230268, V-230271, V-230273, V-230275, V-230276, V-230277, V-230278, V-230279, V-230280, V-230282, V-230286, V-230287, V-230288, V-230290, V-230291, V-230296, V-230298, V-230310, V-230311, V-230312, V-230313, V-230314, V-230315, V-230316, V-230318, V-230319, V-230320, V-230321, V-230322, V-230324, V-230325 V-230326, V-230327, V-230330, V-230333, V-230335, V-230337, V-230339, V-230341, V-230343, V-230345, V-230347, V-230348, V-230352, V-230353, V-230354, V-230356, V-230357, V-230358, V-230359, V-230359, V-230360, V-230361, V-230362, V-230363, V-230365, V-230366, V-230368, V-230369, V-230370, V-230372, V-230373, V-230375, V-230376, V-230377, V-230378, V-230380, V-230382, V-230383, V-230385, V-230386, V-230387, V-230389, V-230390, V-230392, V-230393, V-230394, V-230396, V-230397, V-230398, V-230399, V-230400, V-230401, V-230402, V-230403, V-230404, V-230405, V-230406, V-230407, V-230408, V-230409, V-230410, V-230411, V-230412, V-230413, V-230418, V-230419, V-230421, V-230422, V-230423, V-230424, V-230425, V-230426, V-230427, V-230428, V-230429, V-230430, V-230431, V-230432, V-230433, V-230434, V-230435, V-230436, V-230437, V-230438, V-230439, V-230444, V-230446, V-230447, V-230448, V-230449, V-230455, V-230456, V-230462, V-230463, V-230464, V-230465, V-230466, V-230467, V-230470, V-230471, V-230472, V-230473, V-230474, V-230475, V-230478, V-230480, V-230481, V-230482, V-230483, V-230488, V-230489, V-230493, V-230502, V-230503, V-230505, V-230506, V-230507, V-230523, V-230524, V-230525, V-230526, V-230527, V-230532, V-230535, V-230536, V-230537, V-230538, V-230539, V-230540, V-230541, V-230542, V-230543, V-230544, V-230545 V-230546, V-230547, V-230548, V-230549, V-230550, V-230555, V-230556, V-230557, V-230559, V-230560, V-230561, V-237640, V-237642, V-237643, V-244519, V-244523, V-244524, V-244525, V-244525 44526, V-244528, V-244531, V-244533, V-244535, V-244536,

V-244538, V-244539, V-244542, V-244543, V-244544, V-244547, V-244550, V-244551, V-244552, V-244553, V-24454, V-250315, V-250316, V-250317, V-251707, V-251708, V-251709, V-251710, V-251711, V-251713, V-251714, V-251715, V-251716, V-251717, V-251718, V-256974, V-257258 y V-274877

RHEL 9 STIG versión 2 versión 5

Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de categoría III (baja) de esta distribución de Linux, además de:

- RHEL 9/CentOS Stream 9

V-257780, V-257781, V-257783, V-257786, V-257788, V-257790, V-257791, V-257792, V-257793, V-257794, V-257797, V-257798, V-257799, V-257800, V-257801, V-257802, V-257803, V-257804, V-257805, V-257806, V-257807, V-257808, V-257809, V-257810, V-257811, V-257812, V-257813, V-257814, V-257815, V-257816, V-257817, V-257818, V-257825, V-257827, V-257828, V-257829, V-257830, V-257831, V-257832, V-257833, V-257834, V-257836, V-257838, V-257839, V-257840, V-257841, V-257842, V-257849, V-257882, V-257883, V-257884, V-257885, V-257886, V-257887, V-257888, V-257889, V-257890, V-257891, V-257892, V-257893, V-257894, V-257895, V-257896, V-257897, V-257898, V-257899, V-257900, V-257901, V-257902, V-257903, V-257904, V-257905, V-257906, V-257907, V-257908, V-257909, V-257910, V-257911, V-257912, V-257913, V-257914, V-257915, V-257916, V-257917, V-257918, V-257919, V-257920, V-257921, V-257922, V-257923, V-257924, V-257925, V-257926, V-257927, V-257928, V-257929, V-257930, V-257933, V-257934, V-257935, V-257936, V-257939, V-257940, V-257942, V-257943, V-257944, V-257948, V-257951, V-257952, V-257953, V-257954, V-257957, V-257958, V-257959, V-257960, V-257961, V-257962, V-257963, V-257964, V-257965, V-257966, V-257967, V-257968, V-257969, V-257970, V-257971, V-257972, V-257973, V-257974, V-257975, V-257976, V-257977, V-257978, V-257979, V-257980, V-257982, V-257983, V-257985, V-257987, V-257988, V-257992, V-257993, V-257994, V-257995, V-257996, V-257997, V-257998, V-257999, V-258000, V-258001, V-258002, V-258003, V-258003 8004, V-258005, V-258006, V-258007, V-258008, V-258009, V-258010, V-258011, V-258028, V-258034, V-258035, V-258038, V-258039, V-258040, V-258041, V-258043, V-258046, V-258049, V-258052, V-258054, V-258055, V-258056, V-258057, V-258060, V-258063, V-258064, V-258065, V-258066, V-258068, V-258070, V-258071, V-258072, V-258073, V-258074, V-258075, V-258077, V-258079, V-258080, V-258081, V-258082, V-258083, V-258084, V-258085, V-258088, V-258089, V-258090, V-258091, V-258092, V-258093, V-258095, V-258097, V-258098, V-258099, V-258100, V-258101, V-258102, V-258103, V-258104, V-258105, V-258107, V-258108, V-258109, V-258110, V-258111, V-258112, V-258113, V-258114, V-258115, V-258116, V-258117, V-258118, V-258119,

V-258120, V-258122, V-258123, V-258124, V-258125, V-258126, V-258128, V-258129, V-258130, V-258133, V-258137, V-258140, V-258141, V-258142, V-258144, V-258145, V-258146, V-258147, V-258148, V-258150, V-258151, V-258152, V-258153, V-258154, V-258156, V-258157, V-258158, V-258159, V-258160, V-258161, V-258162, V-258163, V-258164, V-258165, V-258166, V-258167, V-258168, V-258169, V-258170, V-258171, V-258172, V-258175, V-258176, V-258177, V-258178, V-258179, V-258180, V-258181, V-258182, V-258183, V-258184, V-258184 185, V-258186, V-258187, V-258188, V-258189, V-258190, V-258191, V-258192, V-258193, V-258194, V-258195, V-258196, V-258197, V-258198, V-258199, V-258200, V-258201, V-258202, V-258203, V-258204, V-258205, V-258206, V-258207, V-258208, V-258209, V-258210, V-258211, V-258212, V-258213, V-258214, V-258215, V-258216 V-258217, V-258218, V-258219, V-258220, V-258221, V-258222, V-258223, V-258224, V-258225, V-258226, V-258227, V-258228, V-258229, V-258232, V-258233, V-258234, V-258237, V-258239, V-258240, V-270174, V-270175, V-270176, V-270177 V-272488 y V-274878

Amazon Linux 2023 STIG versión 1 versión 1

Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de categoría III (baja) de esta distribución de Linux, además de:

V-273995, V-274000, V-274001, V-274002, V-274003, V-274004, V-274005, V-274006, V-274008, V-274009, V-274010, V-274011, V-274012, V-274013, V-274014, V-274017, V-274018, V-274019, V-274020, V-274020, V-274021, V-274022, V-274023, V-274026, V-274027, V-274028, V-274030, V-274031, V-274032, V-274033, V-274034, V-274035, V-274036, V-274037, V-274040, V-274041, V-274042, V-274044, V-274044 274045, V-274047, V-274048, V-274049, V-274050, V-274051, V-274053, V-274054, V-274059, V-274061, V-274062, V-274065, V-274069, V-274070, V-274071, V-274072, V-274073, V-274074, V-274075, V-274076, V-274077, V-274078, V-274079, V-274081, V-274082, V-274083, V-274084, V-274085, V-274086, V-274087, V-274088, V-274089, V-274090, V-274091, V-274092, V-274093, V-274094, V-274095, V-274096, V-274097, V-274098, V-274099, V-274100, V-274101, V-274102, V-274103, V-274104, V-274105, V-274106, V-274107, V-274108, V-274109, V-274110, V-274111, V-274112, V-274113, V-274114 V-274115, V-274116, V-274117, V-274119, V-274120, V-274121, V-274122, V-274123, V-274124, V-274125, V-274126, V-274127, V-274128, V-274129, V-274130, V-274131, V-274132, V-274133, V-274134, V-274135, V-274136, V-274137, V-274138, V-274139, V-274140, V-274142, V-274143, V-274144, V-274145, V-274147, V-274149, V-274151, V-274152, V-274154, V-274155, V-274156, V-274157, V-274160, V-274161, V-274162, V-274164, V-274165, V-274166, V-274167, V-274168, V-274169, V-274170, V-274173, V-274174, V-274177, V-274181, V-274182, V-274185 y V-274187

SLES 12 STIG versión 3 versión 3

Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de categoría III (baja) de esta distribución de Linux, además de:

V-217102, V-217105, V-217110, V-217116, V-217117, V-217118, V-217119, V-217120, V-217121, V-217122, V-217124, V-217125, V-217126, V-217127, V-217128, V-217130, V-217134, V-217138, V-217143, V-217147, V-217152, V-217153, V-217154, V-217155, V-217156, V-217158, V-217161, V-217163, V-217166, V-217167, V-217168, V-217169, V-217170, V-217171, V-217182, V-217183, V-217188, V-217190, V-217191, V-217194, V-217195, V-217196, V-217197, V-217200, V-217201, V-217202, V-217203, V-217204, V-217205, V-217206, V-217207, V-217208, V-217210, V-217217, V-217218, V-217223, V-217227, V-217230, V-217240, V-217241, V-217242, V-217243, V-217244, V-217245, V-217246, V-217247, V-217248, V-217249, V-217250, V-217251, V-217252, V-217253, V-217254, V-217255, V-217257, V-217258, V-217260, V-217265, V-217266, V-217267, V-217269, V-217272, V-217273, V-217274, V-217275, V-217276, V-217277, V-217278, V-217279, V-217280, V-217283, V-217284, V-217286, V-217287, V-217288 V-217289, V-217290, V-217291, V-217292, V-217293, V-217294, V-217295, V-217296, V-217299, V-217300, V-217301, V-217302, V-233308, V-237606, V-237607, V-237608, V-237609, V-237610, V-237611, V-237612, V-237613, V-237614, V-237615, V-237616, V-237617, V-237618, V-237619, V-237620, V-237621, V-237622, V-237623, V-251720, V-251722, V-255914 y V-256981

SLES 15 STIG, versión 2, versión 5

Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de categoría III (baja) de esta distribución de Linux, además de:

V-234802, V-234807, V-234809, V-234813, V-234815, V-234817, V-234821, V-234822, V-234823, V-234825, V-234827, V-234828, V-234829, V-234830, V-234832, V-234833, V-234834, V-234835, V-234836, V-234837, V-234838, V-234839, V-234840, V-234841, V-234842, V-234843, V-234844, V-234845, V-234848, V-234854, V-234855, V-234856, V-234857, V-234858, V-234861, V-234862, V-234863, V-234869, V-234870, V-234875, V-234880, V-234881, V-234882, V-234883, V-234884, V-234885, V-234886, V-234887, V-234888, V-234889, V-234891, V-234895, V-234896, V-234897, V-234899, V-234900, V-234901, V-234902, V-234903, V-234904, V-234906, V-234910, V-234911, V-234912, V-234913, V-234914, V-234918, V-234924, V-234928, V-234932, V-234937, V-234938, V-234939, V-234940, V-234941, V-234942, V-234943, V-234944, V-234945, V-234946, V-234947, V-234948, V-234949, V-234950, V-234951, V-234952, V-234954, V-234956, V-234957, V-234958, V-234959, V-234961, V-234962, V-234964 V-234966, V-234969, V-234973, V-234975, V-234976, V-234977, V-234978, V-234979, V-234981, V-234982, V-234983, V-234991, V-235002, V-235003, V-235007, V-235008, V-235009, V-235010, V-235013, V-235014, V-235015, V-235016, V-235017,

V-235018, V-235019, V-235020, V-235021, V-235022, V-235023, V-235024, V-235025, V-235026, V-235028, V-235029, V-235030, V-251724, V-255920, V-256983, y V-274879

Ubuntu 18.04 STIG versión 2, versión 15

Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de categoría III (baja) de esta distribución de Linux, además de:

V-219149, V-219155, V-219156, V-219160, V-219166, V-219168, V-219176, V-219181, V-219184, V-219186, V-219188, V-219189, V-219190, V-219191, V-219192, V-219193, V-219194, V-219195, V-219196 V-219197, V-219198, V-219199, V-219200, V-219201, V-219202, V-219203, V-219204, V-219205, V-219206, V-219207, V-219208, V-219209, V-219213, V-219214, V-219215, V-219216, V-219217, V-219218, V-219219, V-219220, V-219221, V-219222, V-219223, V-219224, V-219225, V-219226, V-219227, V-219228, V-219229, V-219230, V-219231, V-219232, V-219233, V-219234, V-219235, V-219236, V-219238, V-219239, V-219240, V-219241, V-219242, V-219243, V-219244, V-219250, V-219254, V-219257, V-219263, V-219264, V-219265, V-219266, V-219267, V-219268, V-219269, V-219270, V-219271, V-219272, V-219273, V-219274, V-219275, V-219276, V-219277, V-219279, V-219281, V-219287, V-219291, V-219296, V-219297, V-219298, V-219299, V-219300, V-219303, V-219304, V-219306, V-219309, V-219310, V-219311, V-219315, V-219318, V-219319, V-219323, V-219326, V-219328, V-219330, V-219331, V-219335, V-219336, V-219337, V-219338, V-219339, V-219342, V-219344, V-233779, V-233780 y V- 255906

Ubuntu 20.04 STIG versión 2, versión 3

Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de categoría III (baja) de esta distribución de Linux, además de:

V-238200, V-238205, V-238207, V-238209, V-238210, V-238211, V-238212, V-238213, V-238220, V-238225, V-238227, V-238228, V-238229, V-238230, V-238231, V-238232, V-238236, V-238238, V-238239, V-238240, V-238241, V-238242 244, V-238245, V-238246, V-238247, V-238248, V-238249, V-238250, V-238251, V-238252, V-238253, V-238254, V-238255, V-238256, V-238257, V-238258, V-238264, V-238268, V-238271, V-238277, V-238278, V-238279, V-238280, V-238281, V-238282, V-238282 238283, V-238284, V-238285, V-238286, V-238287, V-238288, V-238289, V-238290, V-238291, V-238292, V-238293, V-238294, V-238295, V-238297, V-238298, V-238299, V-238300, V-238301, V-238302, V-238303, V-238304, V-238309, V-238310, V-238315, V-238316, V-238317, V-238318, V-238319, V-238320, V-238324, V-238325, V-238329, V-238329 238330, V-238333, V-238334, V-238337, V-238338, V-238339, V-238340, V-238341, V-238342, V-238343, V-238344, V-238345, V-238346, V-238347, V-238348, V-238349, V-238350, V-238351, V-238352,

V-238353, V-238355, V-238356, V-238359, V-238360, V-238369, V-238370, V-238371, V-238376, V-238377, V-238378, V-251505, V-255912, V-274852 y V-274853

Ubuntu 22.04 STIG versión 2 versión 5

Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de categoría III (baja) de esta distribución de Linux, además de:

V-260471, V-260473, V-260474, V-260475, V-260477, V-260478, V-260485, V-260486, V-260487, V-260488, V-260489, V-260490, V-260491, V-260492, V-260493, V-260494, V-260495, V-260496, V-260497, V-260498, V-260499, V-260500, V-260505 V-260506, V-260507, V-260508, V-260509, V-260510, V-260511, V-260512, V-260513, V-260514, V-260522, V-260527, V-260528, V-260530, V-260533, V-260534, V-260535, V-260537, V-260538, V-260540, V-260542, V-260543, V-260545, V-260546, V-260547, V-260553, V-260554, V-260555, V-260556, V-260557, V-260560, V-260561, V-260562, V-260563, V-260564, V-260565, V-260566, V-260567, V-260569, V-260572, V-260573, V-260574, V-260575, V-260576, V-260582, V-260584, V-260585, V-260586, V-260588, V-260589, V-260590, V-260591, V-260594, V-260597, V-260598, V-260599, V-260600, V-260601, V-260602, V-260603, V-260604, V-260605, V-260606, V-260607, V-260608, V-260609, V-260610, V-260611, V-260612, V-260613, V-260614, V-260615, V-260616, V-260617, V-260618, V-260619, V-260620, V-260621, V-260621 0622, V-260623, V-260624, V-260625, V-260626, V-260627, V-260628, V-260629, V-260630, V-260631, V-260632, V-260633, V-260634, V-260635, V-260636, V-260637, V-260638, V-260639, V-260640, V-260641, V-260642, V-260643, V-260644, V-260645, V-260646, V-260647, V-260648, V-260649, V-274862, V-274864 y V-274866

Ubuntu 24.04 STIG versión 1 versión 2

Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de categoría III (baja) de esta distribución de Linux, además de:

V-270649, V-270651, V-270652, V-270653, V-270654, V-270656, V-270657, V-270659, V-270660, V-270661, V-270662, V-270663, V-270669, V-270672, V-270673, V-270674, V-270676, V-270678, V-270679, V-270680, V-270681, V-270683, V-270684, V-270684 270685, V-270686, V-270687, V-270688, V-270689, V-270692, V-270693, V-270696, V-270697, V-270698, V-270699, V-270700, V-270701, V-270702, V-270703, V-270704, V-270705, V-270709, V-270715, V-270716, V-270718, V-270720 270721, V-270722, V-270723, V-270724, V-270725, V-270726, V-270727, V-270728, V-270729, V-270730, V-270731, V-270732, V-270733, V-270737, V-270739, V-270740, V-270741, V-270742, V-270743, V-270746, V-270750, V-270753, V-270755, V-270756, V-270757, V-270758, V-270759, V-270760, V-270765, V-270766, V-270767, V-270768, V-270769, V-270770, V-270771, V-270772, V-270773, V-270775, V-270776, V-270777, V-270778, V-270779, V-270780, V-270781,

V-270782, V-270783, V-270784, V-270785, V-270786, V-270787, V-270788, V-270789, V-270790, V-270791, V-270792, V-270793, V-270794, V-270795, V-270796, V-270797, V-270798, V-270799, V-270800, V-270801, V-270802, V-270803, V-270804, V-270805, V-270806, V-270807, V-270808, V-270809, V-270810, V-270811, V-270812, V-270813, V-270814, V-270815, V-270821, V-270822, V-270823, V-270824, V-270825, V-270826, V-270827, V-270828, V-270829, V-270830, V-270830 31, V-270832, V-274870, V-274871, V-274872 y V-274873

STIG de Linux versión I (Categoría I)

La siguiente lista contiene la configuración de STIG que el componente de endurecimiento aplica a la infraestructura. Si una configuración compatible no es aplicable a la infraestructura, el componente de endurecimiento omite esa configuración y continúa. Por ejemplo, es posible que algunas configuraciones de STIG no se apliquen a los servidores independientes. Las políticas específicas de la organización también pueden afectar la configuración aplicada por el componente de endurecimiento como, por ejemplo, pedir requisitos a los administradores para revisar la configuración de los documentos.

Para obtener una lista completa, consulte la biblioteca de STIGs documentos. Para obtener información acerca de cómo ver la lista completa, consulte Herramientas de visualización de STIG.

> **Note**
>
> Los componentes de protección STIG High de Linux incluyen todas las configuraciones de STIG enumeradas que TOE de AWS se aplican a los componentes de protección STIG Low y Linux STIG Medium, además de las configuraciones de STIG enumeradas que se aplican específicamente a las vulnerabilidades de categoría I.

RHEL 7 STIG, versión 3, versión 15

Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de las categorías II y III (media y baja) de esta distribución de Linux, además de:

- RHEL 7/CentOS 7/ AL2

  V-204424, V-204425, V-204442, V-204443, V-204447, V-204448, V-204455, V-204497, V-204502, V-204594, V-204620 y V-204621

RHEL 8 STIG versión 2 versión 4

Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de las categorías II y III (media y baja) de esta distribución de Linux, además de:

- RHEL 8/CentOS 8

  V-230221, V-230223, V-230226, V-230264, V-230265, V-230283, V-230284, V-230487, V-230492, V-230529, V-230530, V-230531, V-230533, V-230558, V-244540 y V-268322

RHEL 9 STIG versión 2 versión 5

Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de las categorías II y III (media y baja) de esta distribución de Linux, además de:

- RHEL 9/CentOS Stream 9

  V-257784, V-257785, V-257820, V-257821, V-257826, V-257835, V-257955, V-257956, V-257984, V-257986, V-258059, V-258078, V-258094, V-258230, V-258235 y V-258238

Amazon Linux 2023 STIG versión 1 versión 1

Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de las categorías II y III (media y baja) de esta distribución de Linux, además de:

V-273996, V-273997, V-273999, V-274007, V-274038, V-274039, V-274046, V-274052, V-274057 y V-274153

SLES 12 STIG versión 3 versión 3

Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de las categorías II y III (media y baja) de esta distribución de Linux, además de:

V-217101, V-217139, V-217141, V-217142, V-217159, V-217160, V-217264, V-217268, V-222386 y V-251721

SLES 15 STIG versión 2, versión 5

Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de las categorías II y III (media y baja) de esta distribución de Linux, además de:

V-234800, V-234804, V-234818, V-234852, V-234859, V-234860, V-234898, V-234984, V-234985, V-234988, V-234989, V-234990, V-235031, V-235032 y V-251725

Ubuntu 18.04 STIG versión 2, versión 15

Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de las categorías II y III (media y baja) de esta distribución de Linux, además de:

V-219157, V-219158, V-219177, V-219212, V-219308, V-219314, V-219316, V-251507 y V-264388

Ubuntu 20.04 STIG versión 2, versión 3

Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de las categorías II y III (media y baja) de esta distribución de Linux, además de:

V-238201, V-238218, V-238219, V-238326, V-238327, V-238380 y V-251504

Ubuntu 22.04 STIG versión 2, versión 5

Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de las categorías II y III (media y baja) de esta distribución de Linux, además de:

V-260469, V-260482, V-260483, V-260523, V-260524, V-260526, V-260529, V-260539, V-260570, V-260571 y V-260579

Ubuntu 24.04 STIG versión 1, versión 2

Incluye todas las configuraciones de STIG compatibles que el componente de endurecimiento aplica a las vulnerabilidades de las categorías II y III (media y baja) de esta distribución de Linux, además de:

V-270647, V-270648, V-270665, V-270666, V-270708, V-270711, V-270712, V-270713, V-270714, V-270717 y V-270736

Registro del historial de versiones de STIG para Linux

Esta sección registra el historial de versiones de los componentes de Linux. Para ver los cambios y las versiones publicadas de un trimestre, elija el título para ampliar la información.

Cambios en el tercer trimestre de 2025:4 de septiembre de 2025:

Se agregó compatibilidad con el sistema operativo SUSE Linux Enterprise Server (SLES) y Amazon Linux 2023. Se actualizaron las siguientes versiones de STIG y se aplicó el STIGS en la versión del tercer trimestre de 2025 para todos los niveles de conformidad (): low/medium/high

- RHEL 7 STIG, versión 3, versión 15
- RHEL 8 STIG versión 2 versión 4
- RHEL 9 STIG versión 2 versión 5
- Amazon Linux 2023 STIG versión 1 versión 1
- SLES 12 STIG versión 3 versión 3
- SLES 15 STIG versión 2, versión 5
- Ubuntu 18.04 STIG versión 2, versión 15
- Ubuntu 20.04 STIG versión 2, versión 3
- Ubuntu 22.04 STIG versión 2, versión 5
- Ubuntu 24.04 STIG versión 1, versión 2

Cambios en el segundo trimestre de 2025:26 de junio de 2025:

Se actualizaron las siguientes versiones de STIG y se aplicó STIGS para la versión del segundo trimestre de 2025:

STIG-Build-Linux-Low versión 2025.2.x

- RHEL 7 STIG versión 3 versión 15
- RHEL 8 STIG versión 2 versión 3
- RHEL 9 STIG versión 2 versión 4
- Ubuntu 18.04 STIG versión 2, versión 15
- Ubuntu 20.04 STIG versión 2, versión 2
- Ubuntu 22.04 STIG versión 2, versión 4

- Ubuntu 24.04 STIG versión 1, versión 1

STIG-Build-Linux-Medium versión 2025.2.x

- RHEL 7 STIG versión 3 versión 15
- RHEL 8 STIG versión 2 versión 3
- RHEL 9 STIG versión 2 versión 4
- Ubuntu 18.04 STIG versión 2, versión 15
- Ubuntu 20.04 STIG versión 2, versión 2
- Ubuntu 22.04 STIG versión 2, versión 4
- Ubuntu 24.04 STIG versión 1, versión 1

STIG-Build-Linux-High versión 2025.2.x

- RHEL 7 STIG versión 3 versión 15
- RHEL 8 STIG versión 2 versión 3
- RHEL 9 STIG versión 2 versión 4
- Ubuntu 18.04 STIG versión 2, versión 15
- Ubuntu 20.04 STIG versión 2, versión 2
- Ubuntu 22.04 STIG versión 2, versión 4
- Ubuntu 24.04 STIG versión 1, versión 1

Cambios en el primer trimestre de 2025:4 de noviembre de 2025:

Se actualizaron las siguientes versiones de STIG, se aplicó STIGS para la versión del primer trimestre de 2025 y se agregó soporte para Ubuntu 24.04:

STIG-Build-Linux-Low versión 2025.1.x

- RHEL 7 STIG versión 3 versión 15
- RHEL 8 STIG versión 2 versión 2
- RHEL 9 STIG versión 2 versión 3
- Ubuntu 18.04 STIG versión 2, versión 15

- Ubuntu 20.04 STIG versión 2, versión 2
- Ubuntu 22.04 STIG versión 2, versión 3
- Ubuntu 24.04 STIG versión 1, versión 1

STIG-Build-Linux-Medium versión 2025.1.x

- RHEL 7 STIG versión 3 versión 15
- RHEL 8 STIG versión 2 versión 2
- RHEL 9 STIG versión 2 versión 3
- Ubuntu 18.04 STIG versión 2, versión 15
- Ubuntu 20.04 STIG versión 2, versión 2
- Ubuntu 22.04 STIG versión 2, versión 3
- Ubuntu 24.04 STIG versión 1, versión 1

STIG-Build-Linux-High versión 2025.1.x

- RHEL 7 STIG versión 3 versión 15
- RHEL 8 STIG versión 2 versión 2
- RHEL 9 STIG versión 2 versión 3
- Ubuntu 18.04 STIG versión 2, versión 15
- Ubuntu 20.04 STIG versión 2, versión 2
- Ubuntu 22.04 STIG versión 2, versión 3
- Ubuntu 24.04 STIG versión 1, versión 1

Cambios en el cuarto trimestre de 2024:10 de diciembre de 2024:

Se actualizaron las siguientes versiones de STIG, se aplicó el STIGS para la versión del cuarto trimestre de 2024 y se agregó información sobre dos nuevos parámetros de entrada para los componentes de Linux:

STIG-Build-Linux-Low versión 2024.4.x

- RHEL 7 STIG versión 3 versión 15
- RHEL 8 STIG versión 2 versión 1

- RHEL 9 STIG versión 2 versión 2
- Ubuntu 18.04 STIG versión 2, versión 15
- Ubuntu 20.04 STIG versión 2, versión 1
- Ubuntu 22.04 STIG versión 2, versión 2

STIG-Build-Linux-Medium versión 2024.4.x

- RHEL 7 STIG versión 3 versión 15
- RHEL 8 STIG versión 2 versión 1
- RHEL 9 STIG versión 2 versión 2
- Ubuntu 18.04 STIG versión 2, versión 15
- Ubuntu 20.04 STIG versión 2, versión 1
- Ubuntu 22.04 STIG versión 2, versión 2

STIG-Build-Linux-High versión 2024.4.x

- RHEL 7 STIG versión 3 versión 15
- RHEL 8 STIG versión 2 versión 1
- RHEL 9 STIG versión 2 versión 2
- Ubuntu 18.04 STIG versión 2, versión 15
- Ubuntu 20.04 STIG versión 2, versión 1
- Ubuntu 22.04 STIG versión 2, versión 2

Cambios en el tercer trimestre de 2024:4 de octubre de 2024 (sin cambios):

No hubo cambios en el componente de Linux STIGS para la versión del tercer trimestre de 2024.

Cambios en el segundo trimestre de 2024: 10 de mayo de 2024:

Se actualizaron las versiones de STIG y se aplicó el STIGS para la versión del segundo trimestre de 2024. También se ha agregado soporte para RHEL 9, CentOS Stream 9 y Ubuntu 22.04, como sigue:

STIG-Build-Linux-Low versión 2024.2.x

- RHEL 7 STIG versión 3, edición 14

- RHEL 8 STIG versión 1, edición 14
- RHEL 9 STIG versión 1, edición 3
- Ubuntu 18.04 STIG versión 2, edición 14
- Ubuntu 20.04 versión 1, edición 12
- Ubuntu 22.04 versión 1, edición 1

STIG-Build-Linux-Medium versión 2024.2.x

- RHEL 7 STIG versión 3, edición 14
- RHEL 8 STIG versión 1, edición 14
- RHEL 9 STIG versión 1, edición 3
- Ubuntu 18.04 STIG versión 2, edición 14
- Ubuntu 20.04 versión 1, edición 12
- Ubuntu 22.04 versión 1, edición 1

STIG-Build-Linux-High versión 2024.2.x

- RHEL 7 STIG versión 3, edición 14
- RHEL 8 STIG versión 1, edición 14
- RHEL 9 STIG versión 1, edición 3
- Ubuntu 18.04 STIG versión 2, edición 14
- Ubuntu 20.04 versión 1, edición 12
- Ubuntu 22.04 versión 1, edición 1

Cambios en el primer trimestre de 2024 (06/02/2024):

Se actualizaron las versiones de STIG y se aplicó el STIGS para la versión del primer trimestre de 2024 de la siguiente manera:

STIG-Build-Linux-Low versión 2024.1.x

- RHEL 7 STIG versión 3, edición 14
- RHEL 8 STIG versión 1, edición 13

- Ubuntu 18.04 STIG versión 2, edición 13
- Ubuntu 20.04 STIG versión 1, edición 11

STIG-Build-Linux-Medium versión 2024.1.x

- RHEL 7 STIG versión 3, edición 14
- RHEL 8 STIG versión 1, edición 13
- Ubuntu 18.04 STIG versión 2, edición 13
- Ubuntu 20.04 STIG versión 1, edición 11

STIG-Build-Linux-High versión 2024.1.x

- RHEL 7 STIG versión 3, edición 14
- RHEL 8 STIG versión 1, edición 13
- Ubuntu 18.04 STIG versión 2, edición 13
- Ubuntu 20.04 STIG versión 1, edición 11

Cambios en el cuarto trimestre de 2023: 7 de diciembre de 2023:

Se actualizaron las versiones de STIG y se aplicó el STIGS para la versión del cuarto trimestre de 2023 de la siguiente manera:

STIG-Build-Linux-Low versión 2023.4.x

- RHEL 7 STIG versión 3, edición 13
- RHEL 8 STIG versión 1, edición 12
- Ubuntu 18.04 STIG versión 2, edición 12
- Ubuntu 20.04 STIG versión 1, edición 10

STIG-Build-Linux-Medium versión 2023.4.x

- RHEL 7 STIG versión 3, edición 13
- RHEL 8 STIG versión 1, edición 12
- Ubuntu 18.04 STIG versión 2, edición 12

- Ubuntu 20.04 STIG versión 1, edición 10

STIG-Build-Linux-High versión 2023.4.x

- RHEL 7 STIG versión 3, edición 13
- RHEL 8 STIG versión 1, edición 12
- Ubuntu 18.04 STIG versión 2, edición 12
- Ubuntu 20.04 STIG versión 1, edición 10

Cambios en el tercer trimestre de 2023: 4 de octubre de 2023:

Se actualizaron las versiones de STIG y se aplicó el STIGS para la versión del tercer trimestre de 2023 de la siguiente manera:

STIG-Build-Linux-Low versión 2023.3.x

- RHEL 7 STIG versión 3, lanzamiento 12
- RHEL 8 STIG versión 1, lanzamiento 11
- Ubuntu 18.04 STIG versión 2, edición 11
- versión 1 de Ubuntu 20.04 versión 1, lanzamiento 9

STIG-Build-Linux-Medium versión 2023.3.x

- RHEL 7 STIG versión 3, lanzamiento 12
- RHEL 8 STIG versión 1, lanzamiento 11
- Ubuntu 18.04 STIG versión 2, edición 11
- versión 1 de Ubuntu 20.04 versión 1, lanzamiento 9

STIG-Build-Linux-High versión 2023.3.x

- RHEL 7 STIG versión 3, lanzamiento 12
- RHEL 8 STIG versión 1, lanzamiento 11
- Ubuntu 18.04 STIG versión 2, edición 11
- versión 1 de Ubuntu 20.04 versión 1, lanzamiento 9

Cambios en el segundo trimestre de 2023: 3 de mayo de 2023:

Se actualizaron las versiones de STIG y se aplicó el STIGS para la versión del segundo trimestre de 2023 de la siguiente manera:

STIG-Build-Linux-Low versión 2023.2.x

- RHEL 7 STIG versión 3, lanzamiento 11
- RHEL 8 STIG versión 1, lanzamiento 10
- Ubuntu 18.04 STIG versión 2, edición 11
- versión 1 de Ubuntu 20.04 versión 1, lanzamiento 8

STIG-Build-Linux-Medium versión 2023.2.x

- RHEL 7 STIG versión 3, lanzamiento 11
- RHEL 8 STIG versión 1, lanzamiento 10
- Ubuntu 18.04 STIG versión 2, edición 11
- versión 1 de Ubuntu 20.04 versión 1, lanzamiento 8

STIG-Build-Linux-High versión 2023.2.x

- RHEL 7 STIG versión 3, lanzamiento 11
- RHEL 8 STIG versión 1, lanzamiento 10
- Ubuntu 18.04 STIG versión 2, edición 11
- versión 1 de Ubuntu 20.04 versión 1, lanzamiento 8

Cambios en el primer trimestre de 2023: 27 de marzo de 2023:

Se actualizaron las versiones de STIG y se aplicó el STIGS para la versión del primer trimestre de 2023 de la siguiente manera:

STIG-Build-Linux-Low versión 2023.1.x

- RHEL 7 STIG versión 3, lanzamiento 10
- RHEL 8 STIG versión 1, lanzamiento 9
- Ubuntu 18.04 STIG versión 2, edición 10

- versión 1 de Ubuntu 20.04 versión 1, lanzamiento 7

STIG-Build-Linux-Medium versión 2023.1.x

- RHEL 7 STIG versión 3, lanzamiento 10
- RHEL 8 STIG versión 1, lanzamiento 9
- Ubuntu 18.04 STIG versión 2, edición 10
- versión 1 de Ubuntu 20.04 versión 1, lanzamiento 7

STIG-Build-Linux-High versión 2023.1.x

- RHEL 7 STIG versión 3, lanzamiento 10
- RHEL 8 STIG versión 1, lanzamiento 9
- Ubuntu 18.04 STIG versión 2, edición 10
- versión 1 de Ubuntu 20.04 versión 1, lanzamiento 7

Cambios en el cuarto trimestre de 2022: 1 de febrero de 2023:

Se actualizaron las versiones de STIG y se aplicó el STIGS para la versión del cuarto trimestre de 2022 de la siguiente manera:

STIG-Build-Linux-Low versión 2022.4.x

- RHEL 7 STIG versión 3, lanzamiento 9
- RHEL 8 STIG versión 1, lanzamiento 8
- Ubuntu 18.04 STIG versión 2, edición 9
- Ubuntu 20.04 versión 1, lanzamiento 6

STIG-Build-Linux-Medium versión 2022.4.x

- RHEL 7 STIG versión 3, lanzamiento 9
- RHEL 8 STIG versión 1, lanzamiento 8
- Ubuntu 18.04 STIG versión 2, edición 9
- Ubuntu 20.04 versión 1, lanzamiento 6

STIG-Build-Linux-High versión 2022.4.x

- RHEL 7 STIG versión 3, lanzamiento 9
- RHEL 8 STIG versión 1, lanzamiento 8
- Ubuntu 18.04 STIG versión 2, edición 9
- Ubuntu 20.04 versión 1, lanzamiento 6

Cambios en el tercer trimestre de 2022: 30 de septiembre de 2022 (sin cambios):

No se produjeron cambios en el componente STIGS de Linux para la versión del tercer trimestre de 2022.

Cambios en el segundo trimestre de 2022: 2 de agosto de 2022:

Introdujimos el soporte para Ubuntu, actualizamos las versiones de STIG y aplicamos el STIGS para la versión del segundo trimestre de 2022 de la siguiente manera:

STIG-Build-Linux-Low versión 2022.2.x

- RHEL 7 STIG versión 3, lanzamiento 7
- RHEL 8 STIG versión 1, lanzamiento 6
- Ubuntu 18.04 STIG versión 2, versión 6 (nueva)
- Ubuntu 20.04 STIG versión 1, versión 4 (nueva)

STIG-Build-Linux-Medium versión 2022.2.x

- RHEL 7 STIG versión 3, lanzamiento 7
- RHEL 8 STIG versión 1, lanzamiento 6
- Ubuntu 18.04 STIG versión 2, versión 6 (nueva)
- Ubuntu 20.04 STIG versión 1, versión 4 (nueva)

STIG-Build-Linux-High versión 2022.2.x

- RHEL 7 STIG versión 3, lanzamiento 7
- RHEL 8 STIG versión 1, lanzamiento 6
- Ubuntu 18.04 STIG versión 2, versión 6 (nueva)

- Ubuntu 20.04 STIG versión 1, versión 4 (nueva)

Cambios en el primer trimestre de 2022: 26 de abril de 2022:

Refactorizado para incluir un mejor soporte para los contenedores. Combinó el AL2 script anterior con RHEL 7. Se actualizaron las versiones de STIG y se aplicó el STIGS para la versión del primer trimestre de 2022 de la siguiente manera:

STIG-Build-Linux-Low versión 3.6.x

- RHEL 7 STIG versión 3, lanzamiento 6
- RHEL 8 STIG versión 1, lanzamiento 5

STIG-Build-Linux-Medium versión 3.6.x

- RHEL 7 STIG versión 3, lanzamiento 6
- RHEL 8 STIG versión 1, lanzamiento 5

STIG-Build-Linux-High versión 3.6.x

- RHEL 7 STIG versión 3, lanzamiento 6
- RHEL 8 STIG versión 1, lanzamiento 5

Cambios en el cuarto trimestre de 2021: 20 de diciembre de 2021:

Se actualizaron las versiones de STIG y se aplicó el STIGS para la versión del cuarto trimestre de 2021 de la siguiente manera:

STIG-Build-Linux-Low versión 3.5.x

- RHEL 7 STIG versión 3, lanzamiento 5
- RHEL 8 STIG versión 1, lanzamiento 4

STIG-Build-Linux-Medium versión 3.5.x

- RHEL 7 STIG versión 3, lanzamiento 5
- RHEL 8 STIG versión 1, lanzamiento 4

STIG-Build-Linux-High versión 3.5.x

- RHEL 7 STIG versión 3, lanzamiento 5
- RHEL 8 STIG versión 1, lanzamiento 4

Cambios en el tercer trimestre de 2021: 30 de septiembre de 2021:

Se actualizaron las versiones de STIG y se aplicó el STIGS para la versión del tercer trimestre de 2021 de la siguiente manera:

STIG-Build-Linux-Low versión 3.4.x

- RHEL 7 STIG versión 3, lanzamiento 4
- RHEL 8 STIG versión 1, lanzamiento 3

STIG-Build-Linux-Medium versión 3.4.x

- RHEL 7 STIG versión 3, lanzamiento 4
- RHEL 8 STIG versión 1, lanzamiento 3

STIG-Build-Linux-High versión 3.4.x

- RHEL 7 STIG versión 3, lanzamiento 4
- RHEL 8 STIG versión 1, versión 3

Componente validador de cumplimiento del SCAP

El Protocolo de automatización de contenido de seguridad (SCAP) es un conjunto de estándares que los profesionales de TI pueden utilizar para identificar las vulnerabilidades de seguridad de las aplicaciones a fin de garantizar la conformidad. El Comprobador de cumplimiento de SCAP (SCC) es una herramienta de escaneo validada por SCAP y lanzada por el Centro de Asuntos de Guerra Aérea Naval (NIWC) Atlántico. Para obtener más información, consulte Security Content Automation Protocol (SCAP) Compliance Checker (SCC) en el sitio web de NIWC Atlantic.

Los `scap-compliance-checker-linux` componentes TOE de AWS `scap-compliance-checker-windows` y B descargan e instalan el escáner SCC en las instancias de creación y prueba en proceso. Cuando el escáner se ejecuta, realiza escaneos de configuración autenticados

utilizando los puntos de referencia SCAP de DISA y proporciona un informe que incluye la siguiente información. TOE de AWS también escribe la información en los registros de la aplicación.

- Configuración de STIG que se aplica a la instancia.
- Una puntuación de conformidad general de la instancia.

Le recomendamos que ejecute la validación del SCAP como último paso del proceso de compilación, para garantizar que los resultados de la validación de conformidad son precisos.

> **Note**
>
> Puede revisar los informes con una de las herramientas de visualización de STIG. Estas herramientas están disponibles en línea a través de DoD Cyber Exchange.

En las siguientes secciones, se describen los puntos de referencia que incluyen los componentes de validación del SCAP.

scap-compliance-checker-windows versión 2024.03.0

El `scap-compliance-checker-windows` componente se ejecuta en las EC2 instancias que Image Builder crea para crear y probar la imagen. TOE de AWS registra tanto el informe como la puntuación que produce la aplicación SCC.

El componente realiza los siguientes pasos del flujo de trabajo:

1. Descarga e instala la aplicación SCC.
2. Importa los puntos de referencia de conformidad.
3. Ejecuta la validación mediante la aplicación SCC.
4. Guarda el informe de conformidad y la puntuación de forma local en el escritorio de la instancia de compilación.
5. Registra la puntuación de conformidad del informe local en los archivos de registro de la TOE de AWS aplicación.

> **Note**
>
> TOE de AWS actualmente admite la validación de conformidad del SCAP para Windows Server 2012 R2 MS, 2016, 2019 y 2022.

El componente del comprobador de conformidad de SCAP para Windows incluye los siguientes puntos de referencia:

Versión SCC: 5.10

Parámetros de referencia del cuarto trimestre de 2023:

- U_MS_Defender_Antivirus_V2R5_STIG_SCAP_1-2_Benchmark
- U_MS_ _Framework_4-0_V2R2_STIG_SCAP_1-2_benchmark DotNet
- U_MS_ IE11 _V2R6_STIG_SCAP_1-2_benchmark
- U_MS_Windows_2012_y_2012_R2_DC_V3R5_STIG_SCAP_1-2_Benchmark
- U_MS_Windows_2012_y_2012_R2_MS_V3R5_STIG_SCAP_1-2_Benchmark
- U_MS_Windows_Defender_Firewall_V2R3_STIG_SCAP_1-2_benchmark
- U_MS_Windows_Server_2016_V2R7_STIG_SCAP_1-2_benchmark
- U_MS_Windows_Server_2019_V3R2_STIG_SCAP_1-2_benchmark
- U_MS_Windows_Server_2022_V2R2_STIG_SCAP_1-2_benchmark
- U_CAN_Ubuntu_20-04_LTS_V1R10_STIG_SCAP_1-2_benchmark
- U_RHEL_7_V3R15_STIG_SCAP_1-3_benchmark
- U_RHEL_8_V1R13_STIG_SCAP_1-3_benchmark
- U_RHEL_9_V2R1_STIG_SCAP_1-3_benchmark

scap-compliance-checker-linux versión 2021.04.0

El `scap-compliance-checker-linux` componente se ejecuta en las EC2 instancias que Image Builder crea para crear y probar la imagen. TOE de AWS registra tanto el informe como la puntuación que produce la aplicación SCC.

El componente realiza los siguientes pasos del flujo de trabajo:

1. Descarga e instala la aplicación SCC.
2. Importa los puntos de referencia de conformidad.
3. Ejecuta la validación mediante la aplicación SCC.
4. Guarda el informe de conformidad y la puntuación de forma local, en la siguiente ubicación de la instancia de compilación: `/opt/scc/SCCResults`.
5. Registra la puntuación de conformidad del informe local en los archivos de registro de la TOE de AWS aplicación.

> **Note**
>
> TOE de AWS actualmente admite la validación de conformidad del SCAP para RHEL 7/8 y Ubuntu 18.04/20.04. La aplicación SCC actualmente es compatible con la arquitectura x86 para la validación.

El componente del comprobador de conformidad de SCAP para Linux incluye los siguientes puntos de referencia:

Versión SCC: 5.10

Parámetros de referencia del cuarto trimestre de 2023:

- U_CAN_Ubuntu_20-04_LTS_V1R10_STIG_SCAP_1-2_Benchmark
- U_RHEL_7_V3R15_STIG_SCAP_1-3_benchmark
- U_RHEL_8_V1R13_STIG_SCAP_1-3_benchmark
- U_RHEL_9_V2R1_STIG_SCAP_1-3_benchmark
- U_MS_Defender_Antivirus_V2R5_STIG_SCAP_1-2_Benchmark
- U_MS_ _Framework_4-0_V2R2_STIG_SCAP_1-2_benchmark DotNet
- U_MS_ IE11 _V2R6_STIG_SCAP_1-2_benchmark
- U_MS_Windows_2012_y_2012_R2_DC_V3R5_STIG_SCAP_1-2_Benchmark
- U_MS_Windows_2012_y_2012_R2_MS_V3R5_STIG_SCAP_1-2_Benchmark
- U_MS_Windows_Defender_Firewall_V2R3_STIG_SCAP_1-2_benchmark
- U_MS_Windows_Server_2016_V2R7_STIG_SCAP_1-2_benchmark

- U_MS_Windows_Server_2019_V3R2_STIG_SCAP_1-2_benchmark
- U_MS_Windows_Server_2022_V2R2_STIG_SCAP_1-2_benchmark

Historial de versiones de SCAP

En la siguiente tabla se describen cambios importantes en el entorno de SCAP y las configuraciones descritas en este documento.

Cambio	Descripción	Fecha
Actualizaciones del SCAP del primer trimestre de 2025	• scap-compliance-checker-windows versión 2024.03.0 (versión SCC: 5.10) • scap-compliance-checker-windows versión 2025.01.0 (versión SCC: 5.10)	11 de abril de 2025
Actualizaciones de SCAP del cuarto trimestre de 2023	• scap-compliance-checker-windows versión 2023.04.0 (versión SCC: 5.8) • scap-compliance-checker-windows versión 2023.04.0 (versión SCC: 5.8)	20 de diciembre de 2021
Actualizaciones de SCAP del tercer trimestre de 2023	• scap-compliance-checker-windows versión 2023.03.0 (versión SCC: 5.7.2) • scap-compliance-checker-linux versión 2023.03.0 (versión SCC: 5.7.2	13 de noviembre de 2023
Component es de SCAP agregados	Se presentaron los siguientes componentes de SCAP: • Creó la scap-compliance-checker-linux versión 2021.04.0 (versión SCC: 5.4.2) •	20 de diciembre de 2021

Cambio	Descripción	Fecha
	Creó la scap-compliance-checker-linux versión 2021.04.0 (versión SCC: 5.4.2)	

Desarrollo de componentes personalizados para su imagen de Generador de imágenes

Puede crear sus propios componentes para personalizar las imágenes de Generador de imágenes de acuerdo con sus especificaciones exactas. Siga estos pasos para desarrollar componentes personalizados para sus recetas de contenedores o imágenes de Generador de imágenes.

1. Si desea desarrollar el documento de componentes y validarlo localmente, puede instalar la aplicación Ejecutor y orquestador de tareas de AWS(TOE de AWS) y configurarla en su máquina local. Para obtener más información, consulte Configuración manual para desarrollar componentes personalizados con TOE de AWS.
2. Cree un documento de componentes que utilice el marco de documentos de TOE de AWS componentes. Para obtener más información sobre el marco de documentos, consulte Utilice el marco de documentos de TOE de AWS componentes para componentes personalizados.
3. Especifique el documento de componentes al crear un componente personalizado. Para obtener más información, consulte Cree un componente personalizado con Generador de imágenes.

Temas

- Creación de un documento del componente YAML para componentes personalizados en Generador de imágenes
- Cree un componente personalizado con Generador de imágenes

Creación de un documento del componente YAML para componentes personalizados en Generador de imágenes

Para crear un componente, debe proporcionar un documento del componente de la aplicación YAML o JSON. El documento contiene el código que se ejecuta durante las fases y los pasos que defina para personalizar la imagen.

Algunos de los ejemplos de esta sección crean un componente de compilación que llama al módulo de `Update0S` acción en la aplicación de administración de TOE de AWS componentes. El módulo actualiza el sistema operativo. Para obtener más información sobre el módulo de acción `Update0S`, consulte Actualizar OS.

El ejemplo del sistema operativo macOS usa el módulo de acción `ExecuteBash` para instalar y verificar la utilidad `wget`. El módulo de acción `Update0S` no es compatible con macOS. Para obtener más información sobre el módulo de acción `ExecuteBash`, consulte ExecuteBash. Para obtener más información sobre las fases, los pasos y la sintaxis de los documentos del componente de la aplicación TOE de AWS , consulte Cómo usar documentos en TOE de AWS.

> **Note**
>
> Generador de imágenes determina el tipo de componente a partir de las fases definidas en el documento del componente de la siguiente manera:
>
> - Compilación: este es el tipo de componente por defecto. Todo lo que no esté clasificado como componente de prueba es un componente de compilación. Este tipo de componente se ejecuta durante la etapa de compilación de la imagen. Si este componente de compilación tiene una fase de `test` definida, esa fase se ejecuta durante la etapa de prueba.
> - Prueba: para ser considerado un componente de prueba, el documento del componente debe incluir solo una fase, denominada `test`. Para las pruebas relacionadas con las configuraciones de los componentes de compilación, le recomendamos que no utilice un componente de prueba independiente. En su lugar, utilice la fase `test` en el componente de compilación asociado.
>
> Para obtener más información sobre cómo Generador de Imágenes utiliza las etapas y fases para administrar el flujo de trabajo de los componentes en su proceso de compilación, consulte Uso de componentes para personalizar su imagen de Generador de imágenes.

Para crear un documento de componentes de una aplicación YAML para una aplicación de ejemplo, siga los pasos de la pestaña correspondiente al sistema operativo de su imagen.

Linux

Creación de un archivo de componentes YAML

Utilice una herramienta de edición de archivos para crear el documento de componente. Los ejemplos de documentación utilizan un archivo denominado *update-linux-os.yaml*, con el siguiente contenido:

```
# Copyright 2019 Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: MIT-0
#
# Permission is hereby granted, free of charge, to any person obtaining a copy of
 this
# software and associated documentation files (the "Software"), to deal in the
 Software
# without restriction, including without limitation the rights to use, copy, modify,
# merge, publish, distribute, sublicense, and/or sell copies of the Software, and to
# permit persons to whom the Software is furnished to do so.
#
# THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
 IMPLIED,
# INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
# PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
# HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
# OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
# SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
name: update-linux-os
description: Updates Linux with the latest security updates.
schemaVersion: 1
phases:
  - name: build
    steps:
    - name: UpdateOS
      action: UpdateOS
# Document End
```

> **Tip**
>
> Utilice una herramienta como este Validador YAML en línea o una extensión YAML lint en su entorno de código para verificar que su YAML está bien formado.

Windows

Creación de un archivo de componentes YAML

Utilice una herramienta de edición de archivos para crear el documento de componente. Los ejemplos de documentación utilizan un archivo denominado *update-windows-os.yaml*, con el siguiente contenido:

```
# Copyright 2019 Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: MIT-0
#
# Permission is hereby granted, free of charge, to any person obtaining a copy of
 this
# software and associated documentation files (the "Software"), to deal in the
 Software
# without restriction, including without limitation the rights to use, copy, modify,
# merge, publish, distribute, sublicense, and/or sell copies of the Software, and to
# permit persons to whom the Software is furnished to do so.
#
# THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
 IMPLIED,
# INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
# PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
# HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
# OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
# SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
name: update-windows-os
description: Updates Windows with the latest security updates.
schemaVersion: 1.0
phases:
  - name: build
    steps:
      - name: UpdateOS
        action: UpdateOS
# Document End
```

> **Tip**
>
> Utilice una herramienta como este Validador YAML en línea o una extensión YAML lint en su entorno de código para verificar que su YAML está bien formado.

macOS

Creación de un archivo de componentes YAML

Utilice una herramienta de edición de archivos para crear el documento de componente. Los ejemplos de documentación utilizan un archivo denominado *wget-macos.yaml*, con el siguiente contenido:

```
name: WgetInstallDocument
description: This is wget installation document.
schemaVersion: 1.0

phases:
  - name: build
    steps:
      - name: WgetBuildStep
        action: ExecuteBash
        inputs:
          commands:
            - |
              PATH=/usr/local/bin:$PATH
              sudo -u ec2-user brew install wget


  - name: validate
    steps:
      - name: WgetValidateStep
        action: ExecuteBash
        inputs:
          commands:
            - |
              function error_exit {
                echo $1
                echo "{\"failureMessage\":\"$2\"}"
                exit 1
              }

              type wget
              if [ $? -ne 0 ]; then
                error_exit "$stderr" "Wget installation failed!"
              fi

  - name: test
    steps:
      - name: WgetTestStep
        action: ExecuteBash
        inputs:
```

```
        commands:
          - wget -h
```

> **Tip**
>
> Utilice una herramienta como este Validador YAML en línea o una extensión YAML lint en su entorno de código para verificar que su YAML está bien formado.

Cree un componente personalizado con Generador de imágenes

Una vez que haya completado el documento de componentes, puede usarlo para crear un componente personalizado que puedan usar sus recetas de Generador de imágenes. Puede crear un componente personalizado desde la consola de Image Builder, desde la API o SDKs desde la línea de comandos. Para obtener más información sobre cómo crear un componente personalizado con parámetros de entrada y usarlo en sus recetas, consulte Tutorial: creación de un documento del componente con los parámetros de entrada.

En las siguientes secciones se muestra cómo crear componentes desde la consola o la AWS CLI.

Contenido

- Creación de un componente personalizado de la consola
- Cree un componente personalizado a partir del AWS CLI
- Importe un script para crear un componente desde AWS CLI
- Administración automática de versiones de compilación
- Uso de referencias de versión

Creación de un componente personalizado de la consola

Para crear un componente de TOE de AWS aplicación desde la consola de Image Builder, siga estos pasos:

1. Abra la consola EC2 de Image Builder en https://console.aws.amazon.com/imagebuilder/.
2. En el panel de navegación, seleccione Componentes. A continuación, seleccione Crear componente.
3. En la página Crear componente, en Detalles del componente, introduzca lo siguiente:

a. Sistema operativo (SO) de imágenes. Especifique el sistema operativo con el que es compatible el componente.

b. Categoría de componente. En el menú desplegable, seleccione el tipo de componente de compilación o prueba que va a crear.

c. Nombre del componente. Introduzca un nombre para el componente.

d. versión del componente. Introduzca el número de versión del componente.

e. Descripción. Proporcione una descripción opcional que le ayude a identificar el componente.

f. Descripción del cambio. Proporcione una descripción opcional que le ayude a comprender los cambios realizados en esta versión del componente.

4. En la sección Documento de definición, la opción predeterminada es Definir el contenido del documento. El documento del componente define las acciones que Generador de Imágenes realiza en las instancias de compilación y prueba para crear su imagen.

   En el cuadro Contenido, introduzca el contenido del documento de componente YAML. Para empezar con un ejemplo de Hello World para Linux, seleccione la opción Usar ejemplo. Para obtener más información sobre cómo crear un documento de componente YAML o cómo copiar y pegar el ejemplo de UpdateOS de esa página, consulte Creación de un documento del componente YAML para componentes personalizados en Generador de imágenes.

5. Tras introducir los detalles del componente, seleccione Crear componente.

   > **Note**
   >
   > Para ver el nuevo componente al crear o actualizar una receta, aplique el filtro Owned by me a la lista de componentes de compilación o prueba. El filtro se encuentra en la parte superior de la lista de componentes, junto al cuadro de búsqueda.

6. Para eliminar un componente, en la página Componentes, seleccione la casilla al componente que desea eliminar. En el menú desplegable Acciones, seleccione Eliminar componente.

Actualización de un componente

Para crear una nueva versión del componente, siga estos pasos:

1. En función de dónde empiece:

- Desde la página de la lista de componentes: seleccione la casilla junto al nombre del componente y, a continuación, seleccione Crear nueva versión en el menú Acciones.
- En la página de detalles del componente: pulse el botón Crear nueva versión situado en la esquina superior derecha del encabezado.

2. La información del componente ya está rellenada con los valores actuales cuando aparece la página Crear componente. Siga los pasos de creación de un componente para actualizar el componente. Esto garantiza que introduzca una versión semántica única en la versión del componente. Para obtener más información sobre el control de versiones semántico para los recursos de Generador de Imágenes, consulte Control de versiones semántico en Generador de imágenes.

Cree un componente personalizado a partir del AWS CLI

En esta sección, aprenderá a configurar y utilizar los comandos de Image Builder AWS CLI para crear un componente de TOE de AWS aplicación, de la siguiente manera.

- Suba el documento de componente YAML a un bucket de S3 al que pueda hacer referencia desde la línea de comandos.
- Cree el componente de la TOE de AWS aplicación con el create-component comando.
- Enumere las versiones de los componentes con el comando list-components y un filtro de nombres para ver qué versiones ya existen. Puede usar el resultado para determinar cuál debe ser la próxima versión para las actualizaciones.

Para crear un componente de TOE de AWS aplicación a partir de un documento YAML de entrada, siga los pasos que coincidan con la plataforma de sistema operativo de su imagen.

Linux

Almacene el documento del componente de la aplicación en Amazon S3.

Puede usar un bucket de S3 como repositorio para el documento fuente del componente de TOE de AWS la aplicación. Para almacenar el documento del componente, siga estos pasos:

- Cargue del documento en Amazon S3

  Si el documento ocupa menos de 64 KB, puede omitir este paso. Los documentos con un tamaño de 64 KB o más deben almacenarse en Amazon S3.

```
aws s3 cp update-linux-os.yaml s3://amzn-s3-demo-destination-bucket/my-
path/update-linux-os.yaml
```

Creación de un componente a partir del documento YAML

Para simplificar el create-component comando que utiliza en el AWS CLI, cree un archivo JSON que contenga todos los parámetros de los componentes que desee pasar al comando. Incluya la ubicación del documento *update-linux-os.yaml* que creó en los pasos anteriores. El par clave-valor `uri` contiene la referencia del archivo.

> **Note**
>
> La convención de nomenclatura de los valores de datos del archivo JSON sigue el patrón que se especifica para los parámetros de solicitud de operación de la API Image Builder. Para revisar los parámetros de solicitud de comandos de la API, consulte el CreateComponentcomando en la Referencia de la API de EC2 Image Builder.
> Para proporcionar los valores de los datos como parámetros de la línea de comandos, consulte los nombres de los parámetros especificados en la referencia de comandos de la AWS CLI .

1. Creación de un archivo JSON de entrada de la CLI

   Utilice una herramienta de edición de archivos para crear un archivo llamado *create-update-linux-os-component.json*. Incluya el siguiente contenido:

```
{
 "name": "update-linux-os",
 "semanticVersion": "1.1.2",
 "description": "An example component that updates the Linux operating system",
 "changeDescription": "Initial version.",
 "platform": "Linux",
 "uri": "s3://amzn-s3-demo-destination-bucket/my-path/update-linux-os.yaml",
 "kmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/98765432-
b123-456b-7f89-0123456f789c",
 "tags": {
  "MyTagKey-purpose": "security-updates"
 }
```

```
}
```

2. Creación de un componente

   Utilice el siguiente comando para crear el componente, haciendo referencia al nombre del archivo JSON que creó en el paso anterior:

   ```
   aws imagebuilder create-component --cli-input-json file://create-update-linux-os-component.json
   ```

   > **Note**
   >
   > - Debe incluir la notación `file://` al principio de la ruta del archivo JSON.
   > - La ruta del archivo JSON debe seguir la convención apropiada para el sistema operativo base donde se está ejecutando el comando. Por ejemplo, Windows utiliza la barra diagonal inversa (\) para referirse a la ruta de directorios, mientras que Linux y macOS usan la barra diagonal (/).

Windows

Almacene el documento del componente de la aplicación en Amazon S3.

Puede usar un bucket de S3 como repositorio para el documento fuente del componente de la TOE de AWS aplicación. Para almacenar el documento del componente, siga estos pasos:

- Cargue del documento en Amazon S3

  Si el documento ocupa menos de 64 KB, puede omitir este paso. Los documentos con un tamaño de 64 KB o más deben almacenarse en Amazon S3.

  ```
  aws s3 cp update-windows-os.yaml s3://amzn-s3-demo-destination-bucket/my-path/update-windows-os.yaml
  ```

Creación de un componente a partir del documento YAML

Para simplificar el create-component comando que utiliza en el AWS CLI, cree un archivo JSON que contenga todos los parámetros de los componentes que desee pasar al comando. Incluya la

ubicación del documento *update-windows-os.yaml* que creó en los pasos anteriores. El par clave-valor `uri` contiene la referencia del archivo.

> **Note**
>
> La convención de nomenclatura de los valores de datos del archivo JSON sigue el patrón que se especifica para los parámetros de solicitud de operación de la API Image Builder. Para revisar los parámetros de solicitud de comandos de la API, consulte el CreateComponentcomando en la Referencia de la API de EC2 Image Builder.
> Para proporcionar los valores de los datos como parámetros de la línea de comandos, consulte los nombres de los parámetros especificados en la referencia de comandos de la AWS CLI .

1. Creación de un archivo JSON de entrada de la CLI

   Utilice una herramienta de edición de archivos para crear un archivo llamado *create-update-windows-os-component.json*. Incluya el siguiente contenido:

```
{
 "name": "update-windows-os",
 "semanticVersion": "1.1.2",
 "description": "An example component that updates the Windows operating
 system.",
 "changeDescription": "Initial version.",
 "platform": "Windows",
 "uri": "s3://amzn-s3-demo-destination-bucket/my-path/update-windows-os.yaml",
 "kmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/98765432-
b123-456b-7f89-0123456f789c",
 "tags": {
  "MyTagKey-purpose": "security-updates"
 }
}
```

   > **Note**
   >
   > - Debe incluir la notación `file://` al principio de la ruta del archivo JSON.
   > - La ruta del archivo JSON debe seguir la convención apropiada para el sistema operativo base donde se está ejecutando el comando. Por ejemplo, Windows

utiliza la barra diagonal inversa (\) para referirse a la ruta de directorios, mientras que Linux y macOS usan la barra diagonal (/).

2. Creación de un componente

   Utilice el siguiente comando para crear el componente, haciendo referencia al nombre del archivo JSON que creó en el paso anterior:

   ```
   aws imagebuilder create-component --cli-input-json file://create-update-windows-os-component.json
   ```

   Note

   - Debe incluir la notación `file://` al principio de la ruta del archivo JSON.
   - La ruta del archivo JSON debe seguir la convención apropiada para el sistema operativo base donde se está ejecutando el comando. Por ejemplo, Windows utiliza la barra diagonal inversa (\) para referirse a la ruta de directorios, mientras que Linux y macOS usan la barra diagonal (/).

macOS

Almacene el documento del componente de la aplicación en Amazon S3.

Puede usar un bucket de S3 como repositorio para el documento fuente del componente de la TOE de AWS aplicación. Para almacenar el documento del componente, siga estos pasos:

- Cargue del documento en Amazon S3

  Si el documento ocupa menos de 64 KB, puede omitir este paso. Los documentos con un tamaño de 64 KB o más deben almacenarse en Amazon S3.

  ```
  aws s3 cp wget-macos.yaml s3://amzn-s3-demo-destination-bucket/my-path/wget-macos.yaml
  ```

Creación de un componente a partir del documento YAML

Para simplificar el create-component comando que utiliza en el AWS CLI, cree un archivo JSON que contenga todos los parámetros de los componentes que desee pasar al comando. Incluya la ubicación del documento *wget-macos.yaml* que creó en los pasos anteriores. El par clave-valor `uri` contiene la referencia del archivo.

> **Note**
>
> La convención de nomenclatura de los valores de datos del archivo JSON sigue el patrón que se especifica para los parámetros de solicitud de operación de la API Image Builder. Para revisar los parámetros de solicitud de comandos de la API, consulte el CreateComponentcomando en la Referencia de la API de EC2 Image Builder.
> Para proporcionar los valores de los datos como parámetros de la línea de comandos, consulte los nombres de los parámetros especificados en la referencia de comandos de la AWS CLI .

1. Creación de un archivo JSON de entrada de la CLI

   Utilice una herramienta de edición de archivos para crear un archivo llamado *install-wget-macos-component.json*. Incluya el siguiente contenido:

```
{
 "name": "install install-wget-macos-component",
 "semanticVersion": "1.1.2",
 "description": "An example component that installs and verifies the wget
 utility on macOS.",
 "changeDescription": "Initial version.",
 "platform": "macOS",
 "uri": "s3://amzn-s3-demo-destination-bucket/my-path/wget-macos.yaml",
 "kmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/98765432-
b123-456b-7f89-0123456f789c",
 "tags": {
  "MyTagKey-purpose": "install-software"
 }
}
```

2. Creación de un componente

   Utilice el siguiente comando para crear el componente, haciendo referencia al nombre del archivo JSON que creó en el paso anterior:

   ```
   aws imagebuilder create-component --cli-input-json file://install-wget-macos-component.json
   ```

   > **Note**
   >
   > - Debe incluir la notación `file://` al principio de la ruta del archivo JSON.
   > - La ruta del archivo JSON debe seguir la convención apropiada para el sistema operativo base donde se está ejecutando el comando. Por ejemplo, Windows utiliza la barra diagonal inversa (\) para referirse a la ruta de directorios, mientras que Linux y macOS usan la barra diagonal (/).

TOE de AWS el control de versiones de los componentes para las actualizaciones del AWS CLI

TOE de AWS los nombres y las versiones de los componentes están incrustados en el Amazon Resource Name (ARN) del componente, después del prefijo del componente. Cada nueva versión de un componente tiene su propio ARN único. Los pasos para crear una nueva versión son exactamente los mismos que para crear un nuevo componente, siempre que la versión semántica sea única para el nombre de ese componente. Para obtener más información sobre el control de versiones semántico para los recursos de Generador de Imágenes, consulte Control de versiones semántico en Generador de imágenes.

Para asegurarse de asignar la siguiente versión lógica, primero obtenga una lista de las versiones existentes del componente que desee cambiar. Utilice el list-components comando con el nombre AWS CLI y filtre por él.

En este ejemplo, filtre por el nombre del componente que creó en los ejemplos anteriores de Linux. Para mostrar el componente que ha creado, utilice el valor del parámetro `name` del archivo JSON que utilizó en el comando create-component.

```
aws imagebuilder list-components --filters name="name",values="update-linux-os"
{
    "requestId": "123a4567-b890-123c-45d6-ef789ab0cd1e",
```

```
    "componentVersionList": [
        {
            "arn": "arn:aws:imagebuilder:us-west-2:1234560087789012:component/update-
linux-os/1.0.0",
            "name": "update-linux-os",
            "version": "1.0.0",
            "platform": "Linux",
            "type": "BUILD",
            "owner": "123456789012",
            "dateCreated": "2020-09-24T16:58:24.444Z"
        },
        {
            "arn": "arn:aws:imagebuilder:us-west-2:1234560087789012:component/update-
linux-os/1.0.1",
            "name": "update-linux-os",
            "version": "1.0.1",
            "platform": "Linux",
            "type": "BUILD",
            "owner": "123456789012",
            "dateCreated": "2021-07-10T03:38:46.091Z"
        }
    ]
}
```

En función de los resultados, puede determinar cuál debe ser la próxima versión.

Importe un script para crear un componente desde AWS CLI

En algunos casos, puede ser más fácil empezar con un script preexistente. En este caso, puede utilizar el siguiente ejemplo:

En este ejemplo se supone que usted tiene un archivo llamado *import-component.json* (como se muestra). Tenga en cuenta que el archivo hace referencia directamente a un PowerShell script denominado `AdminConfig.ps1` que ya está cargado en*amzn-s3-demo-source-bucket*. Actualmente, `SHELL` es compatible con el componente `format`.

```
{
"name": "MyImportedComponent",
"semanticVersion": "1.0.0",
"description": "An example of how to import a component",
"changeDescription": "First commit message.",
"format": "SHELL",
```

```
"platform": "Windows",
"type": "BUILD",
"uri": "s3://amzn-s3-demo-source-bucket/AdminConfig.ps1",
"kmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/60763706-
b131-418b-8f85-3420912f020c"
}
```

Para crear el componente de un script importado, ejecute el siguiente comando.

```
aws imagebuilder import-component --cli-input-json file://import-component.json
```

Administración automática de versiones de compilación

Al crear un componente con el mismo nombre y la misma versión semántica que un componente existente, Image Builder incrementa automáticamente la versión de compilación (por ejemplo, de `/1` a `/2`, `a/3`, etc.). Esto le permite realizar actualizaciones iterativas de sus componentes sin tener que gestionar manualmente los números de versión, lo que resulta especialmente útil en CI/CD procesos e implementaciones. infrastructure-as-code Si el contenido del componente es idéntico al de la versión de compilación anterior, Image Builder vuelve `ResourceAlreadyExistsException` para evitar que los componentes duplicados consuman la cuota de servicio.

Uso de referencias de versión

Al crear o recuperar un componente, Image Builder proporciona automáticamente patrones de versión preconstruidos ARNs con caracteres comodín en el latestVersionReferences objeto. Estas referencias facilitan el uso de las versiones más recientes de sus componentes en recetas y canalizaciones sin necesidad de analizarlas manualmente. ARNs

Elegir la referencia de versión correcta

- latestVersionArn (x.x.x): utilice siempre la versión más reciente del componente.
- atestMajorVersionArn (1.x.x): utilice las versiones secundarias y de parche más recientes dentro de una versión principal.
- latestMinorVersionArn (1.2.x): utilice únicamente la última versión del parche.
- latestPatchVersionArn (1.2.3): haga referencia a una versión semántica específica, pero obtenga la versión de compilación más reciente.

> **Note**
>
> Para evitar cargos inesperados, asegúrese de limpiar los recursos y canalizaciones que haya creado a partir de los ejemplos de esta guía. Para obtener más información sobre cómo eliminar recursos en Generador de Imágenes, consulte Eliminación de recursos de Generador de imágenes obsoletos o no utilizados.

Cómo utiliza Image Builder la Ejecutor y orquestador de tareas de AWS aplicación para gestionar los componentes

EC2 Image Builder utiliza la aplicación Ejecutor y orquestador de tareas de AWS (TOE de AWS) para organizar flujos de trabajo complejos, modificar las configuraciones del sistema y probar las imágenes sin necesidad de códigos o scripts de DevOps adicionales. Esta aplicación administra y ejecuta los componentes que implementan su esquema de documento declarativo.

TOE de AWS es una aplicación independiente que Image Builder instala en sus instancias de compilación y prueba al crear una imagen. También puede instalarla manualmente en las EC2 instancias para crear sus propios componentes personalizados. No requiere ninguna configuración adicional y también se puede ejecutar en las instalaciones.

Contenido

- TOE de AWS descargas
- Regiones compatibles
- TOE de AWS referencia de comando
- Configuración manual para desarrollar componentes personalizados con TOE de AWS
- Utilice el marco de documentos de TOE de AWS componentes para componentes personalizados
- Módulos de acción compatibles con el administrador de componentes TOE de AWS
- Configurar la entrada para el comando TOE de AWS run

TOE de AWS descargas

Para instalarlo TOE de AWS, elija el enlace de descarga correspondiente a su arquitectura y plataforma. Si se conecta a un punto final de VPC para su servicio (Image Builder, por ejemplo), debe tener una política de punto final personalizada adjunta que incluya el acceso al depósito de S3

para TOE de AWS las descargas. De lo contrario, las instancias de compilación y prueba no podrán descargar el script de arranque (`bootstrap.sh`) ni instalar la TOE de AWS aplicación. Para obtener más información, consulte Creación de una política de puntos de conexión de VPC para Generador de Imágenes.

> **Important**
>
> AWS está eliminando gradualmente la compatibilidad con las versiones 1.0 y 1.1 de TLS. Para acceder al depósito de S3 para realizar TOE de AWS descargas, el software de su cliente debe usar la versión 1.2 o posterior de TLS. Para obtener más información, consulte esta entrada de blog de seguridad de AWS.

Arquitectura	Plataforma	Enlace de descarga	Ejemplo
386	AL 2 y 2023 RHEL 7, 8 y 9 Ubuntu 16.04, 18.04, 20.04, 22.04 y 24.04 CentOS 7 y 8 SUSE 12 y 15	`https://awstoe- <region>.s3.com/latest/linux/386/awstoe`	https://awstoe-us-east-1.s3.us-east-1.amazonaws.com/latest/linux/386/awstoe
AMD64	AL 2 y 2023 RHEL 7, 8 y 9 Ubuntu 16.04, 18.04, 20.04, 22.04 y 24.04 CentOS 7 y 8 Flujo 8 de CentOS SUSE 12 y 15	`https://awstoe- <region>.s3.com/latest/linux/amd64/awstoe`	https://awstoe-us-east-1.s3.us-east-1.amazonaws.com/latest/linux/amd64/awstoe

Arquitectura	Plataforma	Enlace de descarga	Ejemplo
AMD64	macOS 10.14.x (Mojave), 10.15.x (Catalina), 11.x (Big Sur), 12.x (Monterey)	`https://awstoe- region.s3.rs.com/latest/darwin/amd64/awstoe`	https://awstoe-us-east-1.s3.us-east-1.amazonaws.com/latest/darwin/amd64/awstoe
AMD64	Windows Server 2012 R2, 2016, 2019 y 2022	`https://awstoe- <region>.s3s.com/latest/windows/amd64/awstoe.exe`	https://awstoe-us-east-1.s3.us-east-1.amazonaws.com/latest/windows/amd64/awstoe.exe
ARM64	AL 2 y 2023 RHEL 7, 8 y 9 Ubuntu 16.04, 18.04, 20.04, 22.04 y 24.04 CentOS 7 y 8 Flujo 8 de CentOS SUSE 12 y 15	`https://awstoe- <region>.s3s.com/latest/linux/arm64/awstoe`	https://awstoe-us-east-1.s3.us-east-1.amazonaws.com/latest/linux/arm64/awstoe

Regiones compatibles

TOE de AWS se admite como aplicación independiente en las siguientes regiones.

Región de AWS nombre	Región de AWS
Este de EE. UU. (Ohio)	us-east-2
Este de EE. UU. (Norte de Virginia)	us-east-1
AWS GovCloud (Este de EE. UU.)	us-gov-east-1

Región de AWS nombre	Región de AWS
AWS GovCloud (Estados Unidos-Oeste)	us-gov-west-1
Oeste de EE. UU. (Norte de California)	us-west-1
Oeste de EE. UU. (Oregón)	us-west-2
África (Ciudad del Cabo)	af-south-1
Asia-Pacífico (Hong Kong)	ap-east-1
Asia Pacífico (Osaka)	ap-northeast-3
Asia-Pacífico (Seúl)	ap-northeast-2
Asia Pacífico (Mumbai)	ap-south-1
Asia-Pacífico (Hyderabad)	ap-south-2
Asia-Pacífico (Singapur)	ap-southeast-1
Asia-Pacífico (Sídney)	ap-southeast-2
Asia-Pacífico (Yakarta)	ap-southeast-3
Asia-Pacífico (Tokio)	ap-northeast-1
Canadá (centro)	ca-central-1
Europa (Fráncfort)	eu-central-1
Europa (Zúrich)	eu-central-2
Europa (Estocolmo)	eu-north-1
Europa (Milán)	eu-south-1
Europa (España)	eu-south-2
Europa (Irlanda)	eu-west-1

Región de AWS nombre	Región de AWS
Europa (Londres)	eu-west-2
Europa (París)	eu-west-3
Israel (Tel Aviv)	il-central-1
Medio Oriente (EAU)	me-central-1
Medio Oriente (Baréin)	me-south-1
América del Sur (São Paulo)	sa-east-1
China (Pekín)	cn-north-1
China (Ningxia)	cn-northwest-1

TOE de AWS referencia de comando

TOE de AWS es una aplicación de administración de componentes de línea de comandos que se ejecuta en EC2 instancias de Amazon. Cuando Image Builder lanza una instancia de EC2 compilación o prueba, se instala TOE de AWS en la instancia. A continuación, ejecuta TOE de AWS los comandos AWS CLI para instalar o validar los componentes que se especifican en la receta de la imagen o del contenedor.

> **Note**
>
> Algunos módulos de TOE de AWS acción requieren permisos elevados para ejecutarse en un servidor Linux. Para usar permisos elevados, añada el prefijo a la sintaxis del comando con sudo o ejecute el comando sudo su una vez al iniciar sesión antes de ejecutar los comandos que aparecen a continuación. Para obtener más información sobre los módulos de TOE de AWS acción, consulteMódulos de acción compatibles con el administrador de componentes TOE de AWS.

ejecutar

Use el comando run para ejecutar los scripts de documentos de YAML para uno o más documentos componentes.

validar

Ejecute el comando validate para validar la sintaxis del documento YAML de uno o más documentos componentes.

comando awstoe run

Este comando ejecuta los scripts de los documentos de componentes de YAML en el orden en que se incluyen en el archivo de configuración especificado por el parámetro `--config` o en la lista de documentos de componentes especificada por el parámetro `--documents`.

> **Note**
>
> Debe especificar exactamente uno de los siguientes parámetros, nunca ambos:
> --config
> --documents

Sintaxis

```
awstoe run [--config <file path>] [--cw-ignore-failures <?>]
      [--cw-log-group <?>] [--cw-log-region us-west-2] [--cw-log-stream <?>]
      [--document-s3-bucket-owner <owner>] [--documents <file path,file path,...>]
      [--execution-id <?>] [--log-directory <file path>]
      [--log-s3-bucket-name <name>] [--log-s3-bucket-owner <owner>]
      [--log-s3-key-prefix <?>] [--parameters name1=value1,name2=value2...]
      [--phases <phase name>] [--state-directory <directory path>] [--version <?>]
      [--help] [--trace]
```

Parámetros y opciones

Parámetros

--config ***./config-example.json***

Forma abreviada: -c *./config-example.json*

El archivo de configuración (condicional). Este parámetro contiene la ubicación del archivo JSON que contiene los ajustes de configuración de los componentes que ejecuta este comando. Si especifica los ajustes del comando run en un archivo de configuración, no debe especificar el parámetro `--documents`. Para obtener más información sobre la configuración de entrada, consulte Configurar la entrada para el comando TOE de AWS run.

Los campos de ubicación válidos se incluyen:

- Una ruta de archivo local (*`./config-example.json`*)
- Un URI de S3 (`s3://`*`bucket/key`*)

--cw-ignore-failures

Forma abreviada: N/A

Ignore los errores de registro de los CloudWatch registros.

--cw-log-group

Forma abreviada: N/A

El `LogGroup` nombre de los CloudWatch registros.

--cw-log-region

Forma abreviada: N/A

La AWS región que se aplica a los CloudWatch registros.

--cw-log-stream

Forma abreviada: N/A

El `LogStream` nombre de los CloudWatch registros, que indica TOE de AWS dónde transmitir el `console.log` archivo.

--document-s3-bucket-owner

Forma abreviada: N/A

ID de cuenta de propietario del bucket de documentos basados en URI de S3.

--documentos ***`./doc-1.yaml,./doc-n.yaml`***

Forma abreviada: -d *`./doc-1.yaml`*,*`./doc-n`*

Los documentos componentes (condicionales). Este parámetro contiene una lista de ubicaciones de archivos separadas por comas para que se ejecuten los documentos del componente YAML.

Si especifica documentos YAML para el comando run mediante el parámetro `--documents`, no debe especificar el parámetro `--config`.

Los campos de ubicación válidos se incluyen:

- rutas de archivos locales (*`./component-doc-example.yaml`*).
- S3 URIs (`s3://`*`bucket/key`*).
- Versión de compilación del componente Image Builder ARNs (arn:aws:imagebuilder:us-west-:component/ /2021.12.02/1). *`2:123456789012 my-example-component`*

> **Note**
>
> No hay espacios entre los elementos de la lista, solo comas.

--execution-id

Forma abreviada: -i

Este es el identificador único que se aplica a la ejecución del comando run actual. Este identificador se incluye en los nombres de los archivos de salida y de registro para identificarlos de forma exclusiva y vincularlos a la ejecución del comando actual. Si se omite esta configuración, genera TOE de AWS un GUID.

--log-directory

Forma abreviada: -i

El directorio de destino donde se TOE de AWS almacenan todos los archivos de registro de la ejecución de este comando. De forma predeterminada, este directorio está ubicado en el siguiente directorio principal: `TOE_<DATETIME>_<EXECUTIONID>`. Si no especifica el directorio de registro, TOE de AWS utiliza el directorio de trabajo actual (`.`).

--log-s3-nombre de bucket

Forma abreviada: -b

Si los registros de los componentes se almacenan en Amazon S3 (recomendado), TOE de AWS carga los registros de las aplicaciones de los componentes en el bucket de S3 mencionado en este parámetro.

--log-s3-bucket-owner

Forma abreviada: N/A

Si los registros de los componentes se almacenan en Amazon S3 (recomendado), este es el ID de la cuenta del propietario del depósito en el que se TOE de AWS escriben los archivos de registro.

--log-s3-key-prefix

Forma abreviada: -k

Si los registros de los componentes se almacenan en Amazon S3 (recomendado), este es el prefijo de clave de objeto S3 para la ubicación del registro en el bucket.

--parameters ***name1*** =***value1***, ***name2*** =***value2***...

Forma abreviada: N/A

Los parámetros son variables mutables que se definen en el documento del componente, con ajustes que la aplicación que realiza la llamada puede proporcionar en el tiempo de ejecución.

--phases

Forma abreviada: -p

Una lista separada por comas que especifica qué fases se van a ejecutar desde los documentos de los componentes de YAML. Si un documento de componentes incluye fases adicionales, no se ejecutarán.

--directorio de estados

Forma abreviada: -s

La ruta del archivo donde se almacenan los archivos de seguimiento de estado.

--version

Forma abreviada: -v

Especifica la versión de la aplicación del componente.

Opciones

--help

Forma abreviada: -h

Muestra un manual de ayuda para utilizar las opciones de la aplicación de administración de componentes.

--trace

Forma abreviada: -t

Permite el registro detallado en la consola.

comando awstoe validate

Al ejecutar este comando, valida la sintaxis del documento YAML para cada uno de los documentos componentes especificados por el parámetro `--documents`.

Sintaxis

```
awstoe validate [--document-s3-bucket-owner <owner>]
      --documents <file path,file path,...> [--help] [--trace]
```

Parámetros y opciones

Parámetros

--document-s3-bucket-owner

Forma abreviada: N/A

Se proporcionó el identificador de cuenta de origen de los documentos basados en la URI de S3.

--documentos ***./doc-1.yaml,./doc-n.yaml***

Forma abreviada: -d *./doc-1.yaml*,*./doc-n*

Los documentos que lo componen (obligatorios). Este parámetro contiene una lista de ubicaciones de archivos separadas por comas para que se ejecuten los documentos del componente YAML. Los campos de ubicación válidos se incluyen:

- rutas de archivos locales () *./component-doc-example.yaml*
- S3 URIs (`s3://`*bucket/key*)
- Versión de compilación del componente Image Builder ARNs (arn:aws:imagebuilder:us-west-:component/ /2021.12.02/1) *2:123456789012 my-example-component*

> **Note**
>
> No hay espacios entre los elementos de la lista, solo comas.

Opciones

--help

Forma abreviada: -h

Muestra un manual de ayuda para utilizar las opciones de la aplicación de administración de componentes.

--trace

Forma abreviada: -t

Permite el registro detallado en la consola.

Configuración manual para desarrollar componentes personalizados con TOE de AWS

La aplicación Ejecutor y orquestador de tareas de AWS (TOE de AWS) es una aplicación independiente que crea, valida y ejecuta comandos dentro de un marco de definición de componentes. AWS los servicios se pueden utilizar TOE de AWS para organizar los flujos de trabajo, instalar software, modificar las configuraciones del sistema y probar compilaciones de imágenes.

Siga estos pasos para instalar manualmente la TOE de AWS aplicación y utilizarla como una aplicación independiente para desarrollar componentes personalizados. Image Builder se encarga de estos pasos si utiliza la consola o los AWS CLI comandos de Image Builder para crear componentes personalizados. Para obtener más información, consulte Cree componentes personalizados con Generador de imágenes.

Verificar la firma de la descarga de la TOE de AWS instalación

En esta sección se describe el proceso recomendado para comprobar la validez de la descarga de la instalación en sistemas operativos basados TOE de AWS en Linux, macOS y Windows.

Temas

- Verificación de la firma de la descarga de instalación de TOE de AWS en Linux o macOS
- Verificación de la firma de la descarga de instalación de TOE de AWS en Windows

Verificación de la firma de la descarga de instalación de TOE de AWS en Linux o macOS

En este tema se describe el proceso recomendado para comprobar la validez de la descarga de la instalación en sistemas operativos basados TOE de AWS en Linux o macOS.

Siempre que descargue una aplicación de Internet, le recomendamos que autentique la identidad del publicador del software. Además, compruebe que la aplicación no esté alterada o dañada desde su publicación. Esto le protege ante una posible instalación de una versión de la aplicación que contenga un virus u otro código malintencionado.

Si después de seguir los pasos descritos en este tema determina que el software de la aplicación de TOE de AWS ha sido modificado o está dañado, no ejecute el archivo de instalación. En su lugar, póngase en contacto con Soporte Si desea obtener más información sobre sus opciones de soporte, consulte. Soporte

TOE de AWS los archivos de los sistemas operativos basados en Linux y macOS se firman mediante GnuPG una implementación de código abierto del estándar Pretty Good Privacy (OpenPGP) para firmas digitales seguras. GnuPG(también conocido comoGPG) permite comprobar la autenticación y la integridad mediante una firma digital. Amazon EC2 publica una clave pública y firmas que puede usar para verificar las herramientas de Amazon EC2 CLI descargadas. Para obtener más información acerca de PGP y GnuPG (GPG), consulte http://www.gnupg.org.

El primer paso consiste en establecer una relación de confianza con el editor del software. Descargue la clave pública del editor de software, compruebe que el propietario de la clave pública es quien afirma ser y, a continuación, agregue la clave pública a su llavero. Su llavero es una colección de claves públicas conocidas. Tras establecer la autenticidad de la clave pública, puede usarla para verificar la firma de la aplicación.

Temas

- Instalación de las herramientas de la GPG
- Autenticación e importación de la clave pública
- Verificar la firma del paquete

Instalación de las herramientas de la GPG

Si su sistema operativo es Linux, Unix o macOS, las herramientas GPG probablemente ya estarán instaladas. Para comprobar si las herramientas están instaladas en el sistema, escriba gpg en un símbolo del sistema. Si las herramientas de GPG están instaladas, verá un símbolo del sistema de

GPG. Si las herramientas de GPG no están instaladas, verá un mensaje de error que afirma que no se puede encontrar el comando. Puede instalar el paquete GnuPG desde un repositorio.

Para instalar las herramientas de GPG, seleccione el sistema operativo que coincida con su instancia.

Debian-based Linux

En un terminal, ejecute el comando siguiente:

```
apt-get install gnupg
```

Red Hat–based Linux

En un terminal, ejecute el comando siguiente:

```
yum install gnupg
```

macOS

En un terminal, ejecute el comando siguiente:

```
brew install gnupg
```

Autenticación e importación de la clave pública

El siguiente paso del proceso consiste en autenticar la clave TOE de AWS pública y añadirla como clave de confianza a su GPG conjunto de claves.

Para autenticar e importar la clave pública TOE de AWS

1. Obtenga una copia de la clave pública de GPG siguiendo uno de estos métodos:

   - Descargue la clave desde

     https://awstoe - **<region>** .s3. **<region>**.amazonaws. com/assets/awstoe.gpg. Por ejemplo, https://awstoe-us-east-1.s3.us-east-1.amazonaws.com/latest/assets/awstoe.gpg.
   - Copie la clave desde el siguiente texto y péguela en un archivo llamado `awstoe.gpg`. Asegúrese de incluir todo lo que se indica a continuación:

     ```
     -----BEGIN PGP PUBLIC KEY BLOCK-----
     ```

```
Version: GnuPG v2

mQENBF8UqwsBCACdiRF2bkZYaFSDPFC+LIkWLwFvtUCRwAHtD8KIwTJ6LVn3fHAU
GhuK0ZH9mRrqRT2bq/xJjGsnF9VqTj2AJqndGJdDjz75YCZYM+ocZ+r5HSJaeW9i
S5dykHj7Txti2zHe0G5+W0v7v5bPi2sPHsN7XWQ7+G2AMEPTz8PjxY//I0DvMQns
Sle3l9hz6wCClz1l9LbBzTyHfSm5ucTXvNe88XX5Gmt37OCDM7vfli0Ctv8WFoLN
6jbxuA/sV71yIkPm9IYp3+GvaKeT870+sn8/JOOKE/U4sJV1ppbqmuUzDfhrZUaw
8eW8IN9A1FTIuWiZED/5L83UZuQs1S7s2PjlABEBAAG0GkFXU1RPRSA8YXdzdG9l
QGFtYXpvbi5jb20+iQE5BBMBCAAjBQJfFKsLAhsDBwsJCAcDAgEGFQgCCQoLBBYC
AwECHgECF4AACgkQ3r3BVvWuvFJGiwf9EVmrBR77+Qe/DUeXZJYoaFr7If/fVDZl
6V3TC6p0J0Veme7uXleRUTFOjzbh+7e5sDX19HrnPquzCnzfMiqbp4lSoeUuNdOf
FcpuTCQH+M+sIEIgPno4PLl0Uj2uE1o++mxmonBl/Krk+hly8hB2L/9n/vW3L7BN
OMb1Ll9PmgGPbWipcT8KRdz4SUex9TXGYzjlWb3jU3uXetdaQY1M3kVKE1siRsRN
YYDtpcjmwbhjpu4xm19aFqNoAHCDctEsXJA/mkU3erwIRocPyjAZE2dnlkL9ZkFZ
z9DQkcIarbCnybDM5lemBbdhXJ6hezJE/b17VA0t1fY04MoEkn6oJg==
=oyze
-----END PGP PUBLIC KEY BLOCK-----
```

2. En una línea de comandos del directorio en el que guardó awstoe.gpg, utilice el siguiente comando para importar la clave pública a su conjunto de TOE de AWS claves.

```
gpg --import awstoe.gpg
```

El comando devuelve resultados similares a los siguientes:

```
gpg: key F5AEBC52: public key "AWSTOE <awstoe@amazon.com>" imported
gpg: Total number processed: 1
gpg:               imported: 1  (RSA: 1)
```

Anote el valor de clave, lo necesitará en el siguiente paso. En el ejemplo anterior, el valor de la clave es F5AEBC52.

3. Verifique la huella digital ejecutando el siguiente comando, sustituyendo el valor de la clave por el valor del paso anterior:

```
gpg --fingerprint key-value
```

Este comando devuelve resultados similares a los siguientes:

```
pub   2048R/F5AEBC52 2020-07-19
      Key fingerprint = F6DD E01C 869F D639 15E5  5742 DEBD C156 F5AE BC52
```

```
uid       [ unknown] AWSTOE <awstoe@amazon.com>
```

Además, la huella digital debe ser idéntica a la cadena `F6DD E01C 869F D639 15E5 5742 DEBD C156 F5AE BC52` que se muestra en el ejemplo anterior. Compare la huella digital devuelta con la publicada en esta página. Deberían coincidir. Si no coinciden, no instale el script de TOE de AWS instalación y póngase en contacto con nosotros. Soporte

Verificar la firma del paquete

Después instalar las herramientas de `GPG`, autenticar e importar la clave pública de TOE de AWS y comprobar la clave pública es de confianza, estará listo para verificar la firma del script de instalación.

Para verificar la firma del script de instalación de

1. En un símbolo del sistema, ejecute el siguiente comando para descargar el archivo binario de la aplicación:

   ```
   curl -O https://awstoe-<region>.s3.<region>.amazonaws.com/latest/
   linux/<architecture>/awstoe
   ```

   Por ejemplo:

   ```
   curl -O https://awstoe-us-east-1.s3.us-east-1.amazonaws.com/latest/linux/amd64/
   awstoe
   ```

   Los valores admitidos para la **`architecture`** pueden ser `amd64`, `386` y `arm64`.

2. En un símbolo del sistema, ejecute el siguiente comando para descargar el archivo de firma para el archivo binario de aplicación correspondiente desde la misma ruta de prefijo de clave S3:

   ```
   curl -O https://awstoe-<region>.s3.<region>.amazonaws.com/latest/
   linux/<architecture>/awstoe.sig
   ```

   Por ejemplo:

   ```
   curl -O https://awstoe-us-east-1.s3.us-east-1.amazonaws.com/latest/linux/amd64/
   awstoe.sig
   ```

Los valores admitidos para la **`architecture`** pueden ser `amd64`, `386` y `arm64`.

3. Compruebe la firma ejecutando el siguiente comando en una línea de comandos en el directorio donde guardó `awstoe.sig` y en el archivo TOE de AWS de instalación. Ambos archivos deben estar presentes.

```
gpg --verify ./awstoe.sig ~/awstoe
```

El resultado debería tener un aspecto similar al siguiente:

```
gpg: Signature made Mon 20 Jul 2020 08:54:55 AM IST using RSA key ID F5AEBC52
gpg: Good signature from "AWSTOE awstoe@amazon.com" [unknown]
gpg: WARNING: This key is not certified with a trusted signature!
gpg:          There is no indication that the signature belongs to the owner.
Primary key fingerprint: F6DD E01C 869F D639 15E5 5742 DEBD C156 F5AE BC52
```

Si el resultado contiene la expresión `Good signature from "AWSTOE <awstoe@amazon.com>"`, significa que la firma se ha verificado correctamente y que se puede ejecutar el script de instalación de TOE de AWS .

Si el resultado incluye la expresión `BAD signature`, compruebe si ha realizado el procedimiento correctamente. Si sigue recibiendo esta respuesta, no ejecute el archivo de instalación descargado anteriormente y póngase en contacto con Soporte.

A continuación, se describen en detalle las advertencias que podría recibir:

- ADVERTENCIA: Esta clave no está certificada con una firma de confianza. Nada indica que la firma pertenezca al propietario. Lo ideal sería que visitara una AWS oficina y recibiera la clave en persona. Sin embargo, lo habitual es que la descargue desde un sitio web. En este caso, el sitio web es un AWS sitio web.
- gpg: no se han encontrado claves en las que se pueda confiar de forma definitiva. Esto significa que la clave específica no es "definitivamente fiable" para usted (o para otras personas en las que usted confía).

Para obtener más información, consulte http://www.gnupg.org.

Verificación de la firma de la descarga de instalación de TOE de AWS en Windows

En este tema se describe el proceso recomendado para comprobar la validez del archivo de instalación de la Ejecutor y orquestador de tareas de AWS aplicación en sistemas operativos basados en Windows.

Siempre que descargue una aplicación de Internet, le recomendamos que compruebe la identidad del editor del software y verifique que la aplicación no ha sido alterada ni se ha visto corrompida desde que se publicó. Esto le protege ante una posible instalación de una versión de la aplicación que contenga un virus u otro código malintencionado.

Si después de seguir los pasos descritos en este tema determina que el software de la aplicación de TOE de AWS ha sido modificado o está dañado, no ejecute el archivo de instalación. En su lugar, póngase en contacto con. Soporte

Para verificar la validez del archivo binario de awstoe descargado en los sistemas operativos basados en Windows, debe asegurarse de que la huella digital de su certificado de firma de Amazon Services LLC sea igual a este valor:

BA 81 25 EE AC 64 2E A9 F3 C5 93 CA 6D 3E B7 93 7D 68 75 74

> **Note**
>
> Durante el período de despliegue de un nuevo binario, es posible que el certificado de firmante no coincida con la nueva huella digital. Si el certificado del firmante no coincide, verifique que el valor de la huella digital sea:
> F8 83 11 EE F0 4A A2 91 E3 79 21 BA 6B FC AF F8 19 92 12 D7

Para verificar este valor, siga este procedimiento:

1. Haga clic con el botón derecho en el archivo `awstoe.exe` descargado y abra la ventana Properties (Propiedades).
2. Elija la pestaña Firmas digitales.
3. En Signature List, elija Amazon Services LLC y, a continuación, Details.
4. Elija la pestaña General si aún no lo ha hecho, y luego elija Ver certificado.
5. Elija la pestaña Detalles y luego elija Todos en la lista desplegable Mostrar, si aún no está seleccionada.

6. Desplácese hacia abajo hasta que vea el campo Huella digital y, a continuación, seleccione Huella digital. Así se muestra el valor completo de la huella digital en la ventana inferior.

   - Si el valor de la huella digital en la ventana inferior es idéntico a este valor:

     BA 81 25 EE AC 64 2E A9 F3 C5 93 CA 6D 3E B7 93 7D 68 75 74

     entonces el TOE de AWS binario descargado es auténtico y se puede instalar de forma segura.
   - Si el valor de la huella digital en la ventana de detalles inferior no coincide con el valor anterior, no ejecute `awstoe.exe`.

Primeros pasos

- Paso 1: Instalar TOE de AWS
- Paso 2: Defina AWS las credenciales
- Paso 3: desarrollar los documentos de los componentes a nivel local
- Paso 4: Validar TOE de AWS los componentes
- Paso 5: ejecutar TOE de AWS los componentes

Paso 1: Instalar TOE de AWS

Para desarrollar componentes localmente, descargue e instale la TOE de AWS aplicación.

1. Descargue la TOE de AWS aplicación

   Para instalarla TOE de AWS, elija el enlace de descarga adecuado para su arquitectura y plataforma. Para ver la lista completa de los enlaces de descarga de las aplicaciones, consulte TOE de AWS descargas

   > **Important**
   >
   > AWS está eliminando gradualmente la compatibilidad con las versiones 1.0 y 1.1 de TLS. Para acceder al depósito de S3 para realizar TOE de AWS descargas, el software de su cliente debe usar la versión 1.2 o posterior de TLS. Para obtener más información, consulte esta entrada de blog de seguridad de AWS.

2. Verificación de la firma

   Los pasos para verificar la descarga dependen de la plataforma de servidor en la que ejecute la TOE de AWS aplicación después de instalarla. Para verificar la descarga en un servidor Linux, consulte Verificación de la firma en Linux o macOS. Para comprobar la descarga en un servidor Windows, consulte Verificación de la firma en Windows.

> **Note**
>
> TOE de AWS se invoca directamente desde su ubicación de descarga. No es necesario realizar un paso de instalación independiente. Esto también significa que TOE de AWS puede realizar cambios en el entorno local.
> Para asegurarse de aislar los cambios durante el desarrollo de los componentes, le recomendamos que utilice una EC2 instancia para desarrollar y probar TOE de AWS los componentes.

Paso 2: Defina AWS las credenciales

TOE de AWS requiere AWS credenciales para conectarse a otros Servicios de AWS, como Amazon S3 y Amazon CloudWatch, al ejecutar tareas, como:

- Descargar TOE de AWS documentos desde una ruta de Amazon S3 proporcionada por el usuario.
- Ejecutar módulos de acción `S3Download` o `S3Upload`.
- La transmisión de registros a CloudWatch, cuando está habilitada.

Si TOE de AWS ejecutas una EC2 instancia, la ejecución TOE de AWS utiliza los mismos permisos que la función de IAM asociada a la EC2 instancia.

Para obtener más información sobre las funciones de IAM para EC2, consulte Funciones de IAM para Amazon. EC2

Los siguientes ejemplos muestran cómo configurar las AWS credenciales mediante las variables de `AWS_SECRET_ACCESS_KEY` entorno `AWS_ACCESS_KEY_ID` y.

Para establecer estas variables en Linux, MacOS, o Unix, utilice `export`:

```
export AWS_ACCESS_KEY_ID=your_access_key_id
```

```
export AWS_SECRET_ACCESS_KEY=your_secret_access_key
```

Para configurar estas variables en Windows mediante PowerShell, utilice`$env`.

```
$env:AWS_ACCESS_KEY_ID=your_access_key_id
```

```
$env:AWS_SECRET_ACCESS_KEY=your_secret_access_key
```

Para establecer estas variables en Windows con símbolo del sistema, utilice `set`.

```
set AWS_ACCESS_KEY_ID=your_access_key_id
```

```
set AWS_SECRET_ACCESS_KEY=your_secret_access_key
```

Paso 3: desarrollar los documentos de los componentes a nivel local

Los componentes se crean con documentos YAML de texto simple. Para obtener más información sobre sintaxis de documentos, consulte Utilice el marco de documentos de TOE de AWS componentes para componentes personalizados.

Los siguientes son ejemplos de documentos de componentes Hola, mundo que le ayudarán a empezar.

Linux

Algunos de los ejemplos de componentes de Linux de esta guía hacen referencia a un archivo de documento de componentes denominado `hello-world-linux.yml`. Puede utilizar el siguiente documento para empezar con esos ejemplos.

```
name: Hello World
description: This is hello world testing document for Linux.
schemaVersion: 1.0
phases:
  - name: build
    steps:
      - name: HelloWorldStep
        action: ExecuteBash
        inputs:
          commands:
            - echo 'Hello World from the build phase.'
```

```
  - name: validate
    steps:
      - name: HelloWorldStep
        action: ExecuteBash
        inputs:
          commands:
            - echo 'Hello World from the validate phase.'
  - name: test
    steps:
      - name: HelloWorldStep
        action: ExecuteBash
        inputs:
          commands:
            - echo 'Hello World from the test phase.'
```

Windows

Algunos de los ejemplos de componentes de Windows de esta guía hacen referencia a un archivo de documento de componentes denominado `hello-world-windows.yml`. Puede utilizar el siguiente documento para empezar con esos ejemplos.

```
name: Hello World
description: This is Hello World testing document for Windows.
schemaVersion: 1.0
phases:
  - name: build
    steps:
      - name: HelloWorldStep
        action: ExecutePowerShell
        inputs:
          commands:
            - Write-Host 'Hello World from the build phase.'
  - name: validate
    steps:
      - name: HelloWorldStep
        action: ExecutePowerShell
        inputs:
          commands:
            - Write-Host 'Hello World from the validate phase.'
  - name: test
    steps:
      - name: HelloWorldStep
        action: ExecutePowerShell
```

```
        inputs:
          commands:
            - Write-Host 'Hello World from the test phase.'
```

macOS

Algunos de los ejemplos de componentes de macOS de esta guía hacen referencia a un archivo de documento de componentes denominado `hello-world-macos.yml`. Puede utilizar el siguiente documento para empezar con esos ejemplos.

```
name: Hello World
description: This is hello world testing document for macOS.
schemaVersion: 1.0
phases:
  - name: build
    steps:
      - name: HelloWorldStep
        action: ExecuteBash
        inputs:
          commands:
            - echo 'Hello World from the build phase.'
  - name: validate
    steps:
      - name: HelloWorldStep
        action: ExecuteBash
        inputs:
          commands:
            - echo 'Hello World from the validate phase.'
  - name: test
    steps:
      - name: HelloWorldStep
        action: ExecuteBash
        inputs:
          commands:
            - echo 'Hello World from the test phase.'
```

Paso 4: Validar TOE de AWS los componentes

Puede validar la sintaxis de TOE de AWS los componentes localmente con la TOE de AWS aplicación. Los siguientes ejemplos muestran el `validate` comando de la TOE de AWS aplicación para validar la sintaxis de un componente sin ejecutarlo.

> **Note**
>
> La TOE de AWS aplicación solo puede validar la sintaxis del componente para el sistema operativo actual. Por ejemplo, cuando se ejecuta `awstoe.exe` en Windows, no se puede validar la sintaxis de un documento de Linux que utilice el módulo de acción `ExecuteBash`.

Linux o macOS

```
awstoe validate --documents /home/user/hello-world.yml
```

Windows

```
awstoe.exe validate --documents C:\Users\user\Documents\hello-world.yml
```

Paso 5: ejecutar TOE de AWS los componentes

La TOE de AWS aplicación puede ejecutar una o más fases de documentos específicos mediante el argumento de la línea de `--phases` comandos. Los valores admitidos de `--phases` son `build`, `validate` y `test`. Se pueden introducir varios valores de fase como valores separados por comas.

Al proporcionar una lista de fases, la TOE de AWS aplicación ejecuta secuencialmente las fases especificadas de cada documento. Por ejemplo, TOE de AWS ejecuta las `validate` fases `build` y de`document1.yaml`, seguidas de `validate` las fases `build` y de`document2.yaml`.

Para garantizar que sus registros se almacenen de forma segura y se conserven para la solución de problemas, le recomendamos configurar el almacenamiento de registros en Amazon S3. En Generador de Imágenes, la ubicación de Amazon S3 para publicar los registros se especifica en la configuración de infraestructura. Para obtener más información acerca de la configuración de la infraestructura, consulte Administre la configuración de la infraestructura de Generador de imágenes.

Si no se proporciona una lista de fases, la TOE de AWS aplicación ejecuta todas las fases en el orden indicado en el documento YAML.

Para ejecutar fases específicas en uno o varios documentos, utilice los siguientes comandos.

Fase única

```
awstoe run --documents hello-world.yml --phases build
```

Fases múltiples

```
awstoe run --documents hello-world.yml --phases build,test
```

Ejecución de documentos

Ejecución de todas las fases en un solo documento

```
awstoe run --documents documentName.yaml
```

Ejecución de todas las fases en varios documentos

```
awstoe run --documents documentName1.yaml,documentName2.yaml
```

Introduzca la información de Amazon S3 para cargar TOE de AWS los registros desde una ruta local definida por el usuario (recomendado)

```
awstoe run --documents documentName.yaml --log-s3-bucket-name amzn-s3-demo-destination-bucket --log-s3-key-prefix S3KeyPrefix --log-s3-bucket-owner S3BucketOwner --log-directory local_path
```

Ejecución de todas las fases en un solo documento y muestre todos los registros en la consola

```
awstoe run --documents documentName.yaml --trace
```

Comando de ejemplo:

```
awstoe run --documents s3://bucket/key/doc.yaml --phases build,validate
```

Ejecución del documento con un identificador único

```
awstoe run --documents documentName.yaml --execution-id user-provided-id --
phases build,test
```

Obtenga ayuda con TOE de AWS

```
awstoe --help
```

Utilice el marco de documentos de TOE de AWS componentes para componentes personalizados

Para crear un componente mediante el marco de componentes Ejecutor y orquestador de tareas de AWS(TOE de AWS), debes proporcionar un documento basado en YAML que represente las fases y los pasos que se aplican al componente que crees.Servicios de AWS utilice su componente cuando creen una nueva imagen de máquina de Amazon (AMI) o imagen de contenedor.

Temas

- Flujo de trabajo de documentos de componentes
- Registro de componentes
- Encadenamiento de entrada y salida
- Esquema y definiciones del documento
- Ejemplos de documento
- Uso de variables en su documento de componentes personalizados
- Uso de constructos condicionales en TOE de AWS
- Uso de operadores de comparación en los documentos de componentes de TOE de AWS
- Utilice operadores lógicos en los documentos de componentes de TOE de AWS
- Usar constructos en bucle en TOE de AWS

Flujo de trabajo de documentos de componentes

El documento del TOE de AWS componente utiliza fases y pasos para agrupar las tareas relacionadas y organizar esas tareas en un flujo de trabajo lógico para el componente.

> **Tip**
>
> El servicio que usa su componente para crear una imagen puede implementar reglas sobre qué fases usar en su proceso de compilación y cuándo se permite la ejecución de esas fases. Es importante tener esto en cuenta a la hora de diseñar el componente.

Fases

Las fases representan la progresión del flujo de trabajo a lo largo del proceso de compilación de imágenes. Por ejemplo, el servicio Generador de Imágenes utiliza las fases `build` y `validate`

durante la etapa de compilación para las imágenes que produce. Utiliza las fases `test` y `container-host-test` durante la etapa de prueba para garantizar que la instantánea de la imagen o la imagen del contenedor generen los resultados esperados antes de crear la AMI final o distribuir la imagen del contenedor.

Cuando se ejecuta el componente, los comandos asociados a cada fase se aplican en el orden en el que aparecen en el documento del componente.

Reglas para las fases

- Cada nombre de fase debe ser único dentro de un documento.
- Puede definir muchas fases en el documento.
- Debe incluir al menos una de las siguientes fases en su documento:
  - compilación: en el caso de Generador de Imágenes, esta fase se suele utilizar durante la etapa de compilación.
  - validación: en el caso de Generador de Imágenes, esta fase se suele utilizar durante la etapa de compilación.
  - prueba: en el caso de Generador de Imágenes, esta fase se suele utilizar durante la etapa de prueba.
- Las fases siempre se ejecutan en el orden en que están definidas en el documento. El orden en el que se especifican para TOE de AWS los comandos del no AWS CLI tiene ningún efecto.

Steps

Los pasos son unidades de trabajo individuales que definen el flujo de trabajo dentro de cada fase. Los pasos se ejecutan en orden secuencial. Sin embargo, la entrada o salida de un paso también puede introducirse a un paso posterior como entrada. Esto se llama “encadenamiento”.

Reglas para los pasos

- El nombre del paso debe ser único para la fase.
- El paso debe usar una acción compatible (módulo de acción) que devuelva un código de salida.

  Para obtener una lista completa de los módulos de acción compatibles, su funcionamiento, input/output valores y ejemplos, consulteMódulos de acción compatibles con el administrador de componentes TOE de AWS.

Registro de componentes

TOE de AWS crea una nueva carpeta de registro en las EC2 instancias que se utilizan para crear y probar una nueva imagen cada vez que se ejecuta el componente. En el caso de las imágenes del contenedor, la carpeta de registro se almacena en el contenedor.

Para ayudar a solucionar problemas en caso de que algo vaya mal durante el proceso de creación de la imagen, el documento de entrada y todos los archivos de salida que se TOE de AWS crean mientras se ejecuta el componente se almacenan en la carpeta de registro.

El nombre de la carpeta de registro consta de las siguientes partes:

1. Directorio de registro: cuando un servicio ejecuta un TOE de AWS componente, lo transfiere al directorio de registro junto con otras configuraciones del comando. En los siguientes ejemplos, mostramos el formato de archivo de registro que utiliza Generador de Imágenes.
   - Linux y macOS: `/var/lib/amazon/toe/`
   - Windows: `$env:ProgramFiles\Amazon\TaskOrchestratorAndExecutor\`
2. Prefijo del archivo: se trata de un prefijo estándar que se utiliza para todos los componentes: “`TOE_`”.
3. Tiempo de ejecución: es una marca de tiempo en formato YYYY-MM-DD _HH-MM-SS_UTC-0.
4. ID de ejecución: es el GUID que se asigna cuando se ejecutan uno o más componentes.TOE de AWS

Ejemplo: *`/var/lib/amazon/toe/`*`TOE_`*`2021-07-01_12-34-56_UTC-0_a1bcd2e3-45f6-789a-bcde-0fa1b2c3def4`*

TOE de AWS almacena los siguientes archivos principales en la carpeta de registro:

Archivos de entrada

- document.yaml: el documento que se utiliza como entrada para el comando. Una vez ejecutado el componente, este archivo se almacena como un artefacto.

Archivos de salida

- application.log: el registro de la aplicación contiene información a nivel de depuración con marca de tiempo de TOE de AWS sobre lo que ocurre mientras se ejecuta el componente.

- detailedoutput.json: este archivo JSON contiene información detallada sobre el estado de la ejecución, las entradas, las salidas y los errores de todos los documentos, fases y pasos que se aplican al componente a medida que se ejecuta.
- console.log: el registro de la consola contiene toda la información de salida estándar (stdout) y de error estándar (stderr) que TOE de AWS se graba en la consola mientras el componente está en ejecución.
- chaining.json: este archivo JSON representa las optimizaciones que se aplicaron para resolver las expresiones de encadenamiento.TOE de AWS

> **Note**
>
> Es posible que la carpeta de registro también contenga otros archivos temporales que no se incluyen aquí.

Encadenamiento de entrada y salida

La aplicación TOE de AWS de gestión de la configuración proporciona una función para encadenar entradas y salidas mediante la escritura de referencias en los siguientes formatos:

```
{{ phase_name.step_name.inputs/outputs.variable }}
```

o

```
{{ phase_name.step_name.inputs/outputs[index].variable }}
```

La característica de encadenamiento permite reciclar el código y mejorar la capacidad de mantenimiento del documento.

Reglas de encadenamiento

- Las expresiones encadenadas solo se pueden usar en la sección de entradas de cada paso.
- Las declaraciones con expresiones encadenadas deben ir entre comillas. Por ejemplo:
  - Expresión no válida: `echo {{ phase.step.inputs.variable }}`
  - Expresión válida: `"echo {{ phase.step.inputs.variable }}"`
  - Expresión válida: `'echo {{ phase.step.inputs.variable }}'`
- Las expresiones encadenadas pueden hacer referencia a variables de otros pasos y fases del mismo documento. Sin embargo, el servicio que lleva a cabo las llamadas puede tener reglas que

requieran que las expresiones encadenadas funcionen solo en el contexto de una sola etapa. Por ejemplo, Generador de Imágenes no admite el encadenamiento de la etapa de compilación a la etapa de prueba, ya que ejecuta cada etapa de forma independiente.

- Los índices de las expresiones encadenadas siguen la indexación basada en cero. El índice comienza con cero (0) para hacer referencia al primer elemento.

Ejemplos

Para hacer referencia a la variable de origen en la segunda entrada del siguiente paso de ejemplo, el patrón de encadenamiento es `{{ build.SampleS3Download.inputs[1].source }}`.

```
phases:
  - name: 'build'
    steps:
      - name: SampleS3Download
        action: S3Download
        timeoutSeconds: 60
        onFailure: Abort
        maxAttempts: 3
        inputs:
          - source: 's3://sample-bucket/sample1.ps1'
            destination: 'C:\sample1.ps1'
          - source: 's3://sample-bucket/sample2.ps1'
            destination: 'C:\sample2.ps1'
```

Para hacer referencia a la variable de salida (igual a “Hola”) en el siguiente paso de ejemplo, el patrón de encadenamiento es `{{ build.SamplePowerShellStep.outputs.stdout }}`.

```
phases:
  - name: 'build'
    steps:
      - name: SamplePowerShellStep
        action: ExecutePowerShell
        timeoutSeconds: 120
        onFailure: Abort
        maxAttempts: 3
        inputs:
          commands:
            - 'Write-Host "Hello"'
```

Esquema y definiciones del documento

El siguiente es el esquema YAML de un documento.

```
name: (optional)
description: (optional)
schemaVersion: "string"

phases:
  - name: "string"
    steps:
      - name: "string"
        action: "string"
        timeoutSeconds: integer
        onFailure: "Abort|Continue|Ignore"
        maxAttempts: integer
        inputs:
```

Las definiciones de esquema de un documento son las siguientes:

Campo	Description (Descripción)	Tipo	Obligatorio/a
Nombre	Nombre del documento.	Cadena	No
description	Descripción del documento.	Cadena	No
schemaVersion	versión esquemática del documento, actualmente 1.0.	Cadena	Sí
phases	Una lista de fases con sus pasos.	Enumeración	Sí

Las definiciones de esquema de una fase son las siguientes.

Campo	Description (Descripción)	Tipo	Obligatorio/a
Nombre	Nombre de la fase.	Cadena	Sí
pasos	Lista de los pasos de la fase.	Enumeración	Sí

Las definiciones de esquema de un paso son las siguientes.

Campo	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado
Nombre	Nombre definido por el usuario para el paso.	Cadena		
acción	Palabra clave relacionada con el módulo que ejecuta el paso.	Cadena		
timeoutSeconds	Número de segundos que dura el paso antes de fallar o volver a intentar. Además, admite el valor -1, que indica un tiempo de espera infinito. No se admiten valores 0 ni otros valores negativos.	Entero	No	7200 segundos (120 minutos)

Campo	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado
onFailure	Especifica lo que debe hacer el paso en caso de error. Los valores válidos son los siguientes: • Abortar: se produce un error en el paso después del número máximo de intentos y se detiene la ejecución. Establece el estado de la fase y el documento como `Failed`. • Continuar: se produce un error en el paso después del número máximo de intentos y continúa ejecutando los pasos	Cadena	No	Anular

Campo	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado
	restantes. Establece el estado de la fase y el documento como `Failed`. • Ignorar: establece el paso como `IgnoredFailure` después del número máximo de intentos fallidos y continúa ejecutando los pasos restantes. Establece el estado de la fase y el documento como `SuccessWithIgnoredFailure` .			

Campo	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado
maxAttempts	Número máximo de intentos permitidos antes del fallo del paso.	Entero	No	1
inputs	Contiene los parámetros requeridos por el módulo de acción para ejecutar el paso.	Dict	Sí	

Ejemplos de documento

Los siguientes ejemplos muestran los documentos de los AWSTOE componentes que realizan tareas para el sistema operativo de destino.

Linux

Ejemplo 1: ejecución de un archivo binario personalizado

El siguiente es un ejemplo de documento que descarga y ejecuta un archivo binario personalizado en una instancia Linux.

```
name: LinuxBin
description: Download and run a custom Linux binary file.
schemaVersion: 1.0
phases:
  - name: build
    steps:
      - name: Download
        action: S3Download
        inputs:
          - source: s3://<replaceable>amzn-s3-demo-source-bucket</replaceable>/
<replaceable>myapplication</replaceable>
            destination: /tmp/<replaceable>myapplication</replaceable>
```

```
      - name: Enable
        action: ExecuteBash
        onFailure: Continue
        inputs:
          commands:
            - 'chmod u+x {{ build.Download.inputs[0].destination }}'
      - name: Install
        action: ExecuteBinary
        onFailure: Continue
        inputs:
          path: '{{ build.Download.inputs[0].destination }}'
          arguments:
            - '--install'
      - name: Delete
        action: DeleteFile
        inputs:
          - path: '{{ build.Download.inputs[0].destination }}'
```

Windows

Ejemplo 1: instalación de actualizaciones de Windows

El siguiente es un ejemplo de documento que instala todas las actualizaciones disponibles de Windows, ejecuta un script de configuración, valida los cambios antes de crear la AMI y prueba los cambios después de crear la AMI.

```
name: RunConfig_UpdateWindows
description: 'This document will install all available Windows updates and run a
 config script. It will then validate the changes before an AMI is created. Then
 after AMI creation, it will test all the changes.'
schemaVersion: 1.0
phases:
  - name: build
    steps:
      - name: DownloadConfigScript
        action: S3Download
        timeoutSeconds: 60
        onFailure: Abort
        maxAttempts: 3
        inputs:
          - source: 's3://customer-bucket/config.ps1'
            destination: 'C:\config.ps1'
```

```
      - name: RunConfigScript
        action: ExecutePowerShell
        timeoutSeconds: 120
        onFailure: Abort
        maxAttempts: 3
        inputs:
          file: '{{build.DownloadConfigScript.inputs[0].destination}}'

      - name: Cleanup
        action: DeleteFile
        onFailure: Abort
        maxAttempts: 3
        inputs:
          - path: '{{build.DownloadConfigScript.inputs[0].destination}}'

      - name: RebootAfterConfigApplied
        action: Reboot
        inputs:
          delaySeconds: 60

      - name: InstallWindowsUpdates
        action: UpdateOS

  - name: validate
    steps:
      - name: DownloadTestConfigScript
        action: S3Download
        timeoutSeconds: 60
        onFailure: Abort
        maxAttempts: 3
        inputs:
          - source: 's3://customer-bucket/testConfig.ps1'
            destination: 'C:\testConfig.ps1'

      - name: ValidateConfigScript
        action: ExecutePowerShell
        timeoutSeconds: 120
        onFailure: Abort
        maxAttempts: 3
        inputs:
          file: '{{validate.DownloadTestConfigScript.inputs[0].destination}}'

      - name: Cleanup
        action: DeleteFile
```

```
        onFailure: Abort
        maxAttempts: 3
        inputs:
          - path: '{{validate.DownloadTestConfigScript.inputs[0].destination}}'

  - name: test
    steps:
      - name: DownloadTestConfigScript
        action: S3Download
        timeoutSeconds: 60
        onFailure: Abort
        maxAttempts: 3
        inputs:
          - source: 's3://customer-bucket/testConfig.ps1'
            destination: 'C:\testConfig.ps1'

      - name: ValidateConfigScript
        action: ExecutePowerShell
        timeoutSeconds: 120
        onFailure: Abort
        maxAttempts: 3
        inputs:
          file: '{{test.DownloadTestConfigScript.inputs[0].destination}}'
```

Ejemplo 2: Instálelo AWS CLI en una instancia de Windows

El siguiente es un ejemplo de documento que lo instala AWS CLI en una instancia de Windows mediante el archivo de configuración.

```
name: InstallCLISetUp
description: Install &CLI; using the setup file
schemaVersion: 1.0
phases:
  - name: build
    steps:
      - name: Download
        action: S3Download
        inputs:
          - source: s3://aws-cli/AWSCLISetup.exe
            destination: C:\Windows\temp\AWSCLISetup.exe
      - name: Install
        action: ExecuteBinary
        onFailure: Continue
```

```
        inputs:
          path: '{{ build.Download.inputs[0].destination }}'
          arguments:
            - '/install'
            - '/quiet'
            - '/norestart'
      - name: Delete
        action: DeleteFile
        inputs:
          - path: '{{ build.Download.inputs[0].destination }}'
```

Ejemplo 3: Instálelo AWS CLI con el instalador MSI

El siguiente es un ejemplo de documento que lo instala AWS CLI con el instalador MSI.

```
name: InstallCLIMSI
description: Install &CLI; using the MSI installer
schemaVersion: 1.0
phases:
  - name: build
    steps:
      - name: Download
        action: S3Download
        inputs:
          - source: s3://aws-cli/AWSCLI64PY3.msi
            destination: C:\Windows\temp\AWSCLI64PY3.msi
      - name: Install
        action: ExecuteBinary
        onFailure: Continue
        inputs:
          path: 'C:\Windows\System32\msiexec.exe'
          arguments:
            - '/i'
            - '{{ build.Download.inputs[0].destination }}'
            - '/quiet'
            - '/norestart'
      - name: Delete
        action: DeleteFile
        inputs:
          - path: '{{ build.Download.inputs[0].destination }}'
```

macOS

Ejemplo 1: ejecución de un archivo binario de macOS personalizado

El siguiente es un ejemplo de documento que descarga y ejecuta un archivo binario personalizado en una instancia de macOS.

```
name: macOSBin
description: Download and run a binary file on macOS.
schemaVersion: 1.0
phases:
  - name: build
    steps:
      - name: Download
        action: S3Download
        inputs:
          - source: s3://<replaceable>amzn-s3-demo-source-bucket</replaceable>/
<replaceable>myapplication</replaceable>
            destination: /tmp/<replaceable>myapplication</replaceable>
      - name: Enable
        action: ExecuteBash
        onFailure: Continue
        inputs:
          commands:
            - 'chmod u+x {{ build.Download.inputs[0].destination }}'
      - name: Install
        action: ExecuteBinary
        onFailure: Continue
        inputs:
          path: '{{ build.Download.inputs[0].destination }}'
          arguments:
            - '--install'
      - name: Delete
        action: DeleteFile
        inputs:
          - path: '{{ build.Download.inputs[0].destination }}'
```

Uso de variables en su documento de componentes personalizados

Las variables proporcionan una forma de etiquetar los datos con nombres significativos que se pueden usar en toda la aplicación. Puede definir variables personalizadas con formatos sencillos y

legibles para flujos de trabajo complejos y hacer referencia a ellas en el documento de componentes de la aplicación YAML de un componente.TOE de AWS

En esta sección, se proporciona información que te ayudará a definir las variables de tu TOE de AWS componente en el documento de componentes de la aplicación YAML, incluida la sintaxis, las restricciones de nombres y algunos ejemplos.

Constantes

Las constantes son variables inmutables que no se pueden modificar ni anular una vez definidas. Las constantes se pueden definir mediante los valores de la `constants` sección de un TOE de AWS documento.

Reglas de nomenclatura de constantes

- El nombre debe tener entre 3 y 128 caracteres de extensión.
- El nombre solo puede contener caracteres alfanuméricos (a-z, A-Z y 0-9), guiones (-) o guiones bajos (_).
- El nombre debe ser único dentro del documento.
- El nombre debe estar especificado como una cadena YAML.

Sintaxis

```
constants:
  - <name>:
      type: <constant type>
      value: <constant value>
```

Nombre de la clave	Obligatorio	Descripción
`name`	Sí	Nombre de la constante. Debe ser único para el documento (no debe coincidir con ningún otro nombre de parámetro o constante).
`value`	Sí	Valor de la constante.

Nombre de la clave	Obligatorio	Descripción
`type`	Sí	Tipo de la constante. El tipo admitido es `string`.

Valores constantes de referencia en un documento

Puede hacer referencia a las constantes en entradas escalonadas o en bucle dentro de su documento YAML, de la siguiente manera:

- Las referencias constantes distinguen mayúsculas y minúsculas y el nombre debe coincidir exactamente.
- El nombre debe estar entre corchetes dobles. `{{ MyConstant }}`
- Se permiten espacios dentro de las llaves y se recortan automáticamente. Por ejemplo, todas las referencias siguientes son válidas:

  `{{ MyConstant }}`, `{{ MyConstant}}`, `{{MyConstant }}`, `{{MyConstant}}`
- La referencia del documento YAML debe especificarse como una cadena (entre comillas simples o dobles).

  Por ejemplo: `- {{ MyConstant }}` no es válido, ya que no se identifica como una cadena.

  Sin embargo, las siguientes referencias son válidas: `- '{{ MyConstant }}'` y `- "{{ MyConstant }}"`.

Ejemplos

Constante referenciada en las entradas escalonadas

```
name: Download AWS CLI version 2
schemaVersion: 1.0
constants:
  - Source:
      type: string
      value: https://awscli.amazonaws.com/AWSCLIV2.msi
phases:
  - name: build
    steps:
```

```
      - name: Download
        action: WebDownload
        inputs:
          - source: '{{ Source }}'
            destination: 'C:\Windows\Temp\AWSCLIV2.msi'
```

Constante referenciada en las entradas de bucle

```
name: PingHosts
schemaVersion: 1.0
constants:
  - Hosts:
      type: string
      value: 127.0.0.1,amazon.com
phases:
  - name: build
    steps:
      - name: Ping
        action: ExecuteBash
        loop:
          forEach:
            list: '{{ Hosts }}'
            delimiter: ','
        inputs:
          commands:
            - ping -c 4 {{ loop.value }}
```

Parameters

Los parámetros son variables mutables, con ajustes que la aplicación que realiza la llamada puede proporcionar en el tiempo de ejecución. Puede definir los parámetros en la sección `Parameters` del documento YAML.

Reglas para nombres de parámetros

- El nombre debe tener entre 3 y 128 caracteres de extensión.
- El nombre solo puede contener caracteres alfanuméricos (a-z, A-Z y 0-9), guiones (-) o guiones bajos (_).
- El nombre debe ser único dentro del documento.
- El nombre debe estar especificado como una cadena YAML.

Sintaxis

```
parameters:
  - <name>:
      type: <parameter type>
      default: <parameter value>
      description: <parameter description>
```

Nombre de la clave	Obligatorio	Descripción
`name`	Sí	El nombre del parámetro . Debe ser único para el documento (no debe coincidir con ningún otro nombre de parámetro o constante).
`type`	Sí	El tipo de datos del parámetro . Los tipos admitidos incluyen: `string`.
`default`	No	El valor predeterminado para el parámetro.
`description`	No	Describe el parámetro.

Valores de parámetros de referencia en un documento

Puede hacer referencia a los parámetros en entradas escalonadas o en bucle dentro de su documento YAML, de la siguiente manera:

- Las referencias de parámetros distinguen mayúsculas y minúsculas y el nombre debe coincidir exactamente.
- El nombre debe estar entre {{ *MyParameter* }} corchetes dobles.
- Se permiten espacios dentro de las llaves y se recortan automáticamente. Por ejemplo, todas las referencias siguientes son válidas:

  {{ *MyParameter* }}, {{ *MyParameter*}}, {{*MyParameter* }}, {{*MyParameter*}}

- La referencia del documento YAML debe especificarse como una cadena (entre comillas simples o dobles).

  Por ejemplo: `- {{ MyParameter }}` no es válido, ya que no se identifica como una cadena.

  Sin embargo, las siguientes referencias son válidas: `- '{{ MyParameter }}'` y `- "{{ MyParameter }}"`.

Ejemplos

En los siguientes ejemplos, se muestra cómo utilizar los parámetros del documento YAML:

- Consulte un parámetro en las entradas escalonadas:

```
name: Download AWS CLI version 2
schemaVersion: 1.0
parameters:
  - Source:
      type: string
      default: 'https://awscli.amazonaws.com/AWSCLIV2.msi'
      description: The AWS CLI installer source URL.
phases:
  - name: build
    steps:
      - name: Download
        action: WebDownload
        inputs:
          - source: '{{ Source }}'
            destination: 'C:\Windows\Temp\AWSCLIV2.msi'
```

- Consulte un parámetro en las entradas de bucle:

```
name: PingHosts
schemaVersion: 1.0
parameters:
  - Hosts:
      type: string
      default: 127.0.0.1,amazon.com
      description: A comma separated list of hosts to ping.
phases:
  - name: build
    steps:
```

```
    - name: Ping
      action: ExecuteBash
      loop:
        forEach:
          list: '{{ Hosts }}'
          delimiter: ','
      inputs:
        commands:
          - ping -c 4 {{ loop.value }}
```

Anulación de los parámetros en tiempo de ejecución

Puede utilizar la `--parameters` opción AWS CLI con un par clave-valor para establecer el valor de un parámetro en tiempo de ejecución.

- Especifique el par clave-valor del parámetro como nombre y valor, separados por un signo igual (<name>=<value>).
- Varios parámetros deben estar separados por una coma.
- Se ignoran los nombres de los parámetros que no se encuentran en el documento de componentes de YAML.
- Tanto el nombre como el valor del parámetro son obligatorios.

> **Important**
>
> Los parámetros del componente son valores de texto sin formato y se registran en AWS CloudTrail. Le recomendamos que utilice AWS Secrets Manager nuestro almacén de AWS Systems Manager parámetros para almacenar sus secretos. Para obtener más información sobre Secrets Manager, consulte ¿Qué es Secrets Manager? en la Guía del usuario de AWS Secrets Manager. Para obtener más información acerca del almacén de parámetros de AWS Systems Manager, consulte Almacén de parámetros de AWS Systems Manager en la Guía del usuario de AWS Systems Manager.

Sintaxis

```
--parameters name1=value1,name2=value2...
```

Opción de la CLI	Obligatorio	Description (Descripción)
--parámetros *name* =*value*,...	No	Esta opción toma una lista de pares clave-valor, con el nombre del parámetro como clave.

Ejemplos

En los siguientes ejemplos, se muestra cómo utilizar los parámetros del documento YAML:

- El par clave-valor del parámetro especificado en esta opción `--parameter` no es válido:

```
--parameters ntp-server=
```

- Establezca un par clave-valor de un parámetro con la opción `--parameter` en el AWS CLI:

```
--parameters ntp-server=ntp-server-windows-qe.us-east1.amazon.com
```

- Establezca varios pares clave-valor de parámetros con la opción `--parameter` en el AWS CLI:

```
--parameters ntp-server=ntp-server.amazon.com,http-url=https://internal-us-
east1.amazon.com
```

Utilice los parámetros del almacén de parámetros de Systems Manager

Puede hacer referencia a los AWS Systems Manager parámetros del almacén de parámetros (parámetros SSM) en los documentos de sus componentes añadiendo el prefijo a las variables. `aws:ssm` Por ejemplo:

`{{ aws:ssm:/my/param }}`se resuelve en el valor del parámetro SSM. `/my/param`

Esta función admite los siguientes tipos de parámetros de SSM:

- Cadena: se asigna al tipo de TOE de AWS cadena.
- StringList — Se asigna al TOE de AWS`StringList` tipo.
- SecureString — Se asigna al tipo de TOE de AWS cadena.

Para obtener más información sobre el almacén de parámetros, consulte el almacén de AWS Systems Manager parámetros en la Guía del AWS Systems Manager usuario.

También puede hacer referencia a AWS Secrets Manager los secretos mediante un parámetro `SecureString` SSM. Por ejemplo: `{{ aws:ssm:/aws/reference/secretsmanager/test/test-secret }}`. Para obtener más información, consulte Hacer referencia a AWS Secrets Manager los secretos de los parámetros del almacén de parámetros.

> **Important**
>
> Image Builder excluye la resolución de `SecureString` parámetros de sus registros. Sin embargo, también es responsable de garantizar que la información confidencial no se registre mediante los comandos incluidos en el documento del componente. Por ejemplo, si utiliza el `echo` comando con una cadena segura, el comando escribirá un valor de texto simple en el registro.

Permisos de IAM necesarios

Para usar los parámetros de Systems Manager en sus componentes, su rol de instancia debe tener el `ssm:GetParameter` permiso para el ARN del recurso de parámetros. Por ejemplo:

JSON

```
{
 "Version":"2012-10-17",
 "Statement": [
  {
   "Effect": "Allow",
   "Action": "ssm:GetParameter",
   "Resource": "arn:aws:ssm:*:111122223333:parameter/ImageBuilder-*"
  }
 ]
}
```

Para acceder a los valores cifrados, también necesitarás los siguientes permisos:

- `kms:Decrypt`Añada `SecureString` parámetros o AWS Secrets Manager valores cifrados con un administrador por el cliente AWS KMS key.

- `secretsmanager:GetSecretValue`Añádelo si haces referencia a un secreto de Secrets Manager.

Haga referencia a un parámetro SSM en un documento de componentes

El siguiente ejemplo muestra cómo hacer referencia a un parámetro del almacén de parámetros de Systems Manager de los parámetros de Systems Manager de un componente:

```
name: UseSSMParameterVariable
description: This is a sample component document that prints out the value of an SSM
 Parameter. Never do this for a SecureString parameter.
schemaVersion: 1.0

phases:
  - name: verify
    steps:
      - name: EchoParameterValue
        action: ExecuteBash
        inputs:
          commands:
            - echo "Log SSM parameter name: /my/test/param, value {{ aws:ssm:/my/test/
param }}."
```

Resolución dinámica de variables de tiempo de ejecución para los parámetros de SSM

AWSTOE proporciona la siguiente función integrada que puede utilizar en las referencias de variables para manipular o transformar valores en tiempo de ejecución.

función de resolución

La `resolve` función resuelve una referencia de variable dentro de otra referencia de variable, lo que permite hacer referencias dinámicas a nombres de variables. Esto resulta útil cuando se trabaja con parámetros de SSM, en los que parte de la ruta del parámetro puede ser variable y transferirse como parámetro del documento.

La `resolve` función solo admite la resolución dinámica de la parte del nombre de un parámetro de SSM.

Sintaxis

El `dynamic_variable` ejemplo siguiente representa el nombre de un parámetro SSM y debe ser uno de los siguientes:

- Una referencia de parámetro SSM (por ejemplo,) `aws:ssm:/my/param`
- Una referencia de parámetros de un documento de componentes (por ejemplo,*parameter-name*)

```
{{ aws:ssm:resolve(dynamic_variable) }}
```

Ejemplo: resolver un parámetro SSM en tiempo de ejecución

En el siguiente ejemplo, se muestra cómo utilizar la `resolve` función en un documento de componentes de YAML:

```
name: SsmParameterTest
description: This component verifies an SSM parameter variable reference with the echo
 command.
schemaVersion: 1.0

parameters:
  - parameter-name:
      type: string
      description: "test"

phases:
  - name: validate
    steps:
      - name: PrintDynamicVariable
        action: ExecuteBash
        inputs:
          commands:
            - echo "{{ aws:ssm:resolve(parameter-name) }}"
```

Uso de constructos condicionales en TOE de AWS

Los constructos condicionales realizan diferentes acciones en el documento componente en función de si la expresión condicional especificada se evalúa como `true` o `false`. Puede utilizar el constructo `if` para controlar el flujo de ejecución del documento del componente.

Constructo if

Puede utilizar el constructo `if` para evaluar si un paso debe ejecutarse o no. De forma predeterminada, cuando la expresión condicional `if` se evalúa como `true`, TOE de AWS ejecuta el paso y cuando la condición se evalúa como `false`, TOE de AWS se salta el paso. Si se omite un

paso, se considera un paso correcto cuando TOE de AWS evalúa si la fase y el documento se han ejecutado correctamente.

> **Note**
>
> Una instrucción `if` solo se evalúa una vez, incluso si el paso desencadena un reinicio. Si un paso se reinicia, reconoce que la instrucción `if` ya se ha evaluado y continúa donde la dejó.

Sintaxis

```
if:
  - <conditional expression>:
      [then: <step action>]
      [else: <step action>]
```

Nombre de la clave	Obligatorio	Descripción
expresión condicional	Sí	La expresión condicional puede contener exactamente uno de los siguientes tipos de operadores en el nivel superior. • Operador de comparación: para obtener una lista de los operadores de comparación e información sobre su funcionamiento en los documentos de los TOE de AWS componentes, consulteOperadores de comparación. • Operador lógico: los operadores lógicos incluyen `and`, `or` y `not` y funcionan

Nombre de la clave	Obligatorio	Descripción
		con uno o más operadores de comparación. Para obtener más información sobre cómo funcionan los operadores lógicos en los documentos de los componentes de TOE de AWS , consulte Logical operators (Operadores lógicos). Si la expresión debe cumplir varias condiciones, utilice un operador lógico para especificarlas.
then	No	Define la acción que se debe realizar si la expresión condicional se evalúa como `true`.
else	No	Define la acción que se debe realizar si la expresión condicional se evalúa como `false`.

Nombre de la clave	Obligatorio	Descripción
acción de paso	Condicional	Al utilizar `then` o `else`, debe especificar una de las siguientes acciones de pasos: • Abort: TOE de AWS marca el paso como fallido. • Execute: TOE de AWS ejecuta el paso. • Skip: TOE de AWS omite el paso.

Ejemplo 1: instalación de un paquete

Los siguientes pasos de ejemplo de un documento de TOE de AWS componentes utilizan operadores lógicos para probar el valor de un parámetro y ejecutar los comandos del administrador de paquetes adecuados para instalar una aplicación si el paquete está descomprimido.

```
- name: InstallUnzipAptGet
  action: ExecuteBash
  if:
    and:
        - binaryExists: 'apt-get'
        - not:
            binaryExists: 'unzip'
  inputs:
    commands:
        - sudo apt-get update
        - sudo apt-get install -y unzip

- name: InstallUnzipYum
  action: ExecuteBash
  if:
    and:
        - binaryExists: 'yum'
```

```
            - not:
                binaryExists: 'unzip'
      inputs:
        commands:
            - sudo yum install -y unzip

    - name: InstallUnzipZypper
      action: ExecuteBash
      if:
        and:
            - binaryExists: 'zypper'
            - not:
                binaryExists: 'unzip'
      inputs:
        commands:
            - sudo zypper refresh
            - sudo zypper install -y unzip
```

Ejemplo 2: omisión de un paso

El siguiente ejemplo muestra dos formas de omitir un paso. Una usa un operador lógico y la otra usa un operador de comparación con la acción del paso `Skip`.

```
# Creates a file if it does not exist using not
- name: CreateMyConfigFile-1
  action: ExecuteBash
  if:
    not:
      fileExists: '/etc/my_config'
  inputs:
    commands:
      - echo "Hello world" > '/etc/my_config'

# Creates a file if it does not exist using then and else
- name: CreateMyConfigFile-2
  action: ExecuteBash
  if:
    fileExists: '/etc/my_config'
    then: Skip
    else: Execute
  inputs:
    commands:
      - echo "Hello world" > '/etc/my_config'
```

Uso de operadores de comparación en los documentos de componentes de TOE de AWS

Puede utilizar los siguientes operadores de comparación con el módulo de acción Assert y con las expresiones condicionales que utilizan Constructo if. Un operador de comparación puede operar con un único valor, por ejemplo, `stringIsEmpty`, o puede comparar un valor de referencia con un segundo valor (valor variable) para determinar si la expresión condicional se evalúa como `true` o `false`.

Si la comparación funciona con dos valores, el segundo valor puede ser una variable encadenante.

Al comparar valores de un tipo diferente, se pueden producir las siguientes conversiones de valores antes de la comparación:

- Para las comparaciones numéricas, si el valor de la variable es una cadena, TOE de AWS convierte la cadena en un número antes de la evaluación. Si la conversión no es posible, la comparación devuelve el resultado `false`. Por ejemplo, si el valor de la variable es `"1.0"`, la conversión funciona, pero si el valor de la variable es `"a10"`, la conversión falla.
- Para las comparaciones de cadenas, si el valor de la variable es un número, lo TOE de AWS convierte en una cadena antes de la evaluación.

Comparación de cadenas

Los siguientes operadores de comparación funcionan con cadenas para comparar valores, comprobar si hay espacios o una cadena vacía, o para comparar un valor de entrada con un patrón de expresiones regulares. Las comparaciones de cadenas no distinguen entre mayúsculas y minúsculas y no recortan los espacios desde el principio o el final de las entradas de cadenas.

Operadores de comparación de cadenas

- stringIsEmpty
- stringIsWhitespace
- stringEquals
- stringLessThan
- stringLessThanEquals
- stringGreaterThan

- stringGreaterThanEquals
- patternMatches

stringIsEmpty

El operador `stringIsEmpty` devuelve `true` si la cadena especificada no contiene ningún carácter. Por ejemplo:

```
# Evaluates to true
stringIsEmpty: ""

# Evaluates to false
stringIsEmpty: " "

# Evaluates to false
stringIsEmpty: "Hello."
```

stringIsWhitespace

Comprueba si la cadena especificada para `stringIsWhitespace` contiene solo espacios. Por ejemplo:

```
# Evaluates to true
stringIsWhitespace: "   "

# Evaluates to false
stringIsWhitespace: ""

# Evaluates to false
stringIsWhitespace: " Hello?"
```

stringEquals

Comprueba si la cadena especificada para `stringEquals` coincide exactamente con la cadena especificada en el parámetro `value`. Por ejemplo:

```
# Evaluates to true
stringEquals: 'Testing, testing...'
value: 'Testing, testing...'
```

```
# Evaluates to false
stringEquals: 'Testing, testing...'
value: 'Hello again.'

# Evaluates to false
stringEquals: 'Testing, testing...'
value: 'TESTING, TESTING....'

# Evaluates to false
stringEquals: 'Testing, testing...'
value: '   Testing, testing...'

# Evaluates to false
stringEquals: 'Testing, testing...'
value: 'Testing, testing...   '
```

stringLessThan

Comprueba si la cadena especificada para `stringLessThan` es inferior a la cadena especificada en el parámetro `value`. Por ejemplo:

```
# Evaluates to true
# This comparison operator isn't case sensitive
stringlessThan: 'A'
value: 'a'

# Evaluates to true - 'a' is less than 'b'
stringlessThan: 'b'
value: 'a'

# Evaluates to true
# Numeric strings compare as less than alphabetic strings
stringlessThan: 'a'
value: '0'

# Evaluates to false
stringlessThan: '0'
value: 'a'
```

stringLessThanEs igual a

Comprueba si la cadena especificada para `stringLessThanEquals` es inferior o igual a la cadena especificada en el parámetro `value`. Por ejemplo:

```
# Evaluates to true - 'a' is equal to 'a'
stringLessThanEquals: 'a'
value: 'a'

# Evaluates to true - since the comparison isn't case sensitive, 'a' is equal to 'A'
stringLessThanEquals: 'A'
value: 'a'

# Evaluates to true - 'a' is less than 'b'
stringLessThanEquals: 'b'
value: 'a'

# Evaluates to true - '0' is less than 'a'
stringLessThanEquals: 'a'
value: '0'

# Evaluates to false - 'a' is greater than '0'
stringLessThanEquals: '0'
value: 'a'
```

stringGreaterThan

Comprueba si la cadena especificada para `stringGreaterThan` es superior a la cadena especificada en el parámetro `value`. Por ejemplo:

```
# Evaluates to false - since the comparison isn't case sensitive, 'A' is equal to
 'a'
stringGreaterThan: 'a'
value: 'A'

# Evaluates to true - 'b' is greater than 'a'
stringGreaterThan: 'a'
value: 'b'

# Evaluates to true - 'a' is greater than '0'
stringGreaterThan: '0'
value: 'a'

# Evaluates to false - '0' is less than 'a'
stringGreaterThan: 'a'
value: '0'
```

stringGreaterThanIguales

Comprueba si la cadena especificada para `stringGreaterThanEquals` es superior o igual a la cadena especificada en el parámetro `value`. Por ejemplo:

```
# Evaluates to true - 'a' is equal to 'A'
stringGreaterThanEquals: 'A'
value: 'a'

# Evaluates to true - 'b' is greater than 'a'
stringGreaterThanEquals: 'a'
value: 'b'

# Evaluates to true - 'a' is greater than '0'
stringGreaterThanEquals: '0'
value: 'a'

# Evaluates to false - '0' is less than 'a'
stringGreaterThanEquals: 'a'
value: '0'
```

patternMatches

Comprueba si la cadena especificada en el parámetro `value` coincide con el patrón de expresiones regulares especificado para `patternMatches`. La comparación utiliza el paquete de expresiones regulares de Golang, que se ajusta a la sintaxis. RE2 Para obtener más información sobre RE2 las reglas, consulte el repositorio google/ re2 en. GitHub

En el siguiente ejemplo, se muestra una coincidencia de patrones que devuelve `true`:

```
patternMatches: '^[a-z]+$'
value: 'ThisIsValue'
```

Comparación de números

Los siguientes operadores de comparación funcionan con números. Los valores proporcionados para estos operadores deben ser de uno de los siguientes tipos, según la especificación YAML. La admisión de comparaciones numéricas utiliza el operador de comparación de paquetes grandes de Golang, por ejemplo: func (*Float) Cmp.

- Entero

- Float (basado en float64, admite números del -1,7e+308 al +1,7e+308)
- Cadena que coincide con el siguiente patrón de expresiones regulares: `^[-+]?([0-9]+[.])?[0-9]+$`

Operadores de comparación de números

- numberEquals
- numberLessThan
- numberLessThanEquals
- numberGreaterThan
- numberGreaterThanEquals

numberEquals

Comprueba si el número especificado para `numberEquals` es igual al número especificado en el parámetro `value`. Todos los ejemplos de comparación siguientes devuelven `true`:

```
# Values provided as a positive number
numberEquals: 1
value: 1

# Comparison value provided as a string
numberEquals: '1'
value: 1

# Value provided as a string
numberEquals: 1
value: '1'

# Values provided as floats
numberEquals: 5.0
value: 5.0

# Values provided as a negative number
numberEquals: -1
value: -1
```

numberLessThan

Comprueba si el número especificado para `numberLessThan` es inferior al número especificado en el parámetro `value`. Por ejemplo:

```
# Evaluates to true
numberLessThan: 2
value: 1

# Evaluates to true
numberLessThan: 2
value: 1.9

# Evaluates to false
numberLessThan: 2
value: '2'
```

numberLessThanEs igual a

Comprueba si el número especificado para `numberLessThanEquals` es inferior o igual al número especificado en el parámetro `value`. Por ejemplo:

```
# Evaluates to true
numberLessThanEquals: 2
value: 1

# Evaluates to true
numberLessThanEquals: 2
value: 1.9

# Evaluates to true
numberLessThanEquals: 2
value: '2'

# Evaluates to false
numberLessThanEquals: 2
value: 2.1
```

numberGreaterThan

Comprueba si el número especificado para `numberGreaterThan` es superior al número especificado en el parámetro `value`. Por ejemplo:

```
# Evaluates to true
numberGreaterThan: 1
value: 2

# Evaluates to true
numberGreaterThan: 1
value: 1.1

# Evaluates to false
numberGreaterThan: 1
value: '1'
```

numberGreaterThanIguales

Comprueba si el número especificado para `numberGreaterThanEquals` es superior o igual al número especificado en el parámetro `value`. Por ejemplo:

```
# Evaluates to true
numberGreaterThanEquals: 1
value: 2

# Evaluates to true
numberGreaterThanEquals: 1
value: 1.1

# Evaluates to true
numberGreaterThanEquals: 1
value: '1'

# Evaluates to false
numberGreaterThanEquals: 1
value: 0.8
```

Control de archivos

Los siguientes operadores de comparación comprueban el hash del archivo o comprueban si existe un archivo o una carpeta.

Operadores de archivos y carpetas

- binaryExists

- fileExists
- folderExists
- fileMD5Equals
- fileSHA1Equals
- fileSHA256Equals
- fileSHA512Equals

binaryExists

Comprueba si una aplicación está disponible en la ruta actual. Por ejemplo:

```
binaryExists: 'foo'
```

> **Note**
>
> En los sistemas Linux y macOS, para una aplicación llamada*foo*, esto funciona igual que el siguiente comando bash:type ***foo*** >/dev/null 2>&1, donde $? == 0 indica que la comparación se ha realizado correctamente.
> En los sistemas Windows, para una aplicación denominada*foo*, funciona igual que el PowerShell comando que $LASTEXITCODE = 0 indica que la comparación & C:\Windows\System32\where.exe /Q ***foo*** se ha realizado correctamente.

fileExists

Comprueba si un archivo existe en la ruta especificada. Puede proporcionar una ruta absoluta o relativa. Si la ubicación que especifique existe y es un archivo, la comparación se evaluará como `true`. Por ejemplo:

```
fileExists: '/path/to/file'
```

> **Note**
>
> En los sistemas Linux y macOS, funciona igual que el siguiente comando bash: -d ***/path/to/file***, donde $? == 0 indica que la comparación se ha realizado correctamente.

En los sistemas Windows, funciona igual que el PowerShell comandoTest-Path -Path '***C:\path\to\file***' -PathType 'Leaf'.

folderExists

Comprueba si existe una carpeta en la ruta especificada. Puede proporcionar una ruta absoluta o relativa. Si la ubicación que especifique existe y es una carpeta, la comparación se evaluará como `true`. Por ejemplo:

```
folderExists: '/path/to/folder'
```

> **Note**
>
> En los sistemas Linux y macOS, funciona igual que el siguiente comando bash: -d ***/path/to/folder***, donde $? == 0 indica que la comparación se ha realizado correctamente.
> En los sistemas Windows, funciona igual que el PowerShell comandoTest-Path -Path '***C:\path\to\folder***' -PathType 'Container'.

el archivo MD5 es igual a

Comprueba si el MD5 hash de un archivo es igual a un valor especificado. Por ejemplo:

```
fileMD5Equals: '<MD5Hash>'
path: '/path/to/file'
```

el archivo SHA1 es igual a

Comprueba si el SHA1 hash de un archivo es igual a un valor especificado. Por ejemplo:

```
fileSHA1Equals: '<SHA1Hash>'
path: '/path/to/file'
```

el archivo SHA256 es igual a

Comprueba si el SHA256 hash de un archivo es igual a un valor especificado. Por ejemplo:

```
fileSHA256Equals: '<SHA256Hash>'
```

```
path: '/path/to/file'
```

el archivo SHA512 es igual a

Comprueba si el SHA512 hash de un archivo es igual a un valor especificado. Por ejemplo:

```
fileSHA512Equals: '<SHA512Hash>'
path: '/path/to/file'
```

Utilice operadores lógicos en los documentos de componentes de TOE de AWS

Puede utilizar los siguientes operadores lógicos para añadir o modificar expresiones condicionales en el documento componente. TOE de AWS evalúa las expresiones condicionales en el orden en que se especifican las condiciones. Para obtener más información sobre los operadores de comparación de los documentos de componentes, consulte Uso de operadores de comparación en los documentos de componentes de TOE de AWS.

y

Con el operador `and`, puede evaluar dos o más comparaciones como una sola expresión. La expresión se evalúa como `true` cuando se cumplen todas las condiciones de la lista. De lo contrario, la expresión se evalúa como `false`.

Ejemplos:

En el siguiente ejemplo se realizan dos comparaciones: una cadena y un número. Ambas comparaciones son verdaderas, por lo que la expresión se evalúa como verdadera.

```
and:
  - stringEquals: 'test_string'
    value: 'test_string'
  - numberEquals: 1
    value: 1
```

En el siguiente ejemplo también se realizan dos comparaciones. La primera comparación es falsa, momento en el que se detiene la evaluación y se omite la segunda comparación. La expresión se evalúa como `false`.

```
and:
  - stringEquals: 'test_string'
```

```
    value: 'Hello world!'
  - numberEquals: 1
    value: 1
```

o bien

Con el operador `or`, puede evaluar dos o más comparaciones como una sola expresión. La expresión se evalúa como `true` cuando una de las comparaciones especificadas es verdadera. Si ninguna de las comparaciones especificadas se evalúa como `true`, la expresión se evalúa como `false`.

Ejemplos:

En el siguiente ejemplo se realizan dos comparaciones: una cadena y un número. La primera comparación es verdadera, por lo que la expresión se evalúa como `true` y se omite la segunda comparación.

```
or:
  - stringEquals: 'test_string'
    value: 'test_string'
  - numberEquals: 1
    value: 3
```

En el siguiente ejemplo también se realizan dos comparaciones. La primera comparación es falsa y la evaluación continúa. La segunda comparación es verdadera, por lo que la expresión se evalúa como `true`.

```
or:
  - stringEquals: 'test_string'
    value: 'Hello world!'
  - numberEquals: 1
    value: 1
```

En el último ejemplo, ambas comparaciones son falsas, por lo que la expresión se evalúa como `false`.

```
or:
  - stringEquals: 'test_string'
    value: 'Hello world!'
  - numberEquals: 1
    value: 3
```

not

Con el operador `not`, puede anular una sola comparación. La expresión se evalúa como `true` si la comparación es falsa. Si la comparación es verdadera, la expresión se evalúa como `false`.

Ejemplos:

En el siguiente ejemplo, se realiza una comparación de cadenas. La comparación es falsa, por lo que la expresión se evalúa como `true`.

```
not:
  - stringEquals: 'test_string'
    value: 'Hello world!'
```

En el siguiente ejemplo también se realiza una comparación de cadenas. La comparación es verdadera, por lo que la expresión se evalúa como `false`.

```
not:
  - stringEquals: 'test_string'
    value: 'test_string'
```

Usar constructos en bucle en TOE de AWS

En esta sección se proporciona información que le ayudará a crear constructos en bucle en TOE de AWS. Los constructos en bucle definen una secuencia repetida de instrucciones. Puede utilizar los siguientes tipos de constructos en bucle en TOE de AWS:

- Constructos `for`: se repiten en iteraciones sobre una secuencia acotada de números enteros.
- Constructos `forEach`
  - Bucle `forEach` con lista de entradas: se repite en iteraciones sobre una colección finita de cadenas.
  - Bucle `forEach` con lista delimitada: repite una colección finita de cadenas unidas por un delimitador.

> **Note**
>
> Los constructos en bucle solo admiten tipos de datos de cadena.

Temas de constructos en bucle

- Variables de iteración de referencia
- Tipos de constructos en bucle
- Campos de pasos
- Salidas de paso e iteración

Variables de iteración de referencia

Para hacer referencia al índice y al valor de la variable de iteración actual, la expresión de referencia `{{ loop.* }}` debe usarse en el cuerpo de entrada de un paso que contenga un constructo en bucle. Esta expresión no se puede utilizar para hacer referencia a las variables de iteración del constructo en bucle de otro paso.

La expresión de referencia consta de los siguientes miembros:

- `{{ loop.index }}`: la posición ordinal de la iteración actual, que está indexada en `0`.
- `{{ loop.value }}`: el valor asociado a la variable de iteración actual.

Nombres de bucles

Todos los constructos en bucle tienen un campo de nombre opcional para su identificación. Si se proporciona un nombre de bucle, este se puede utilizar para hacer referencia a las variables de iteración del cuerpo de entrada del paso. Para hacer referencia a los índices y valores de iteración de un bucle denominado, utilice `{{ <loop_name>.* }}` con `{{ loop.* }}` en el cuerpo de entrada del paso. Esta expresión no se puede utilizar para hacer referencia al constructo en bucle denominado de otro paso.

La expresión de referencia consta de los siguientes miembros:

- `{{ <loop_name>.index }}`: la posición ordinal de la iteración actual del bucle denominado, que está indexada en `0`.
- `{{ <loop_name>.value }}` : el valor asociado a la variable de iteración actual del bucle denominado.

Resolución de expresiones de referencia

TOE de AWS Resuelve las expresiones de referencia de la siguiente manera:

- `{{ <loop_name>.* }}`— TOE de AWS resuelve esta expresión mediante la siguiente lógica:
  - Si el bucle del paso que se está ejecutando actualmente coincide con el valor `<loop_name>`, la expresión de referencia se convierte en el constructo en bucle del paso que se está ejecutando actualmente.
  - `<loop_name>` se resuelve en el constructo de bucle denominado si aparece dentro del paso que se está ejecutando actualmente.
- `{{ loop.* }}`— TOE de AWS resuelve la expresión mediante la construcción de bucle definida en el paso que se está ejecutando actualmente.

Si se utilizan expresiones de referencia en un paso que no contiene un bucle, TOE de AWS no resuelve las expresiones y aparecen en el paso sin sustituirlas.

> **Note**
>
> Las expresiones de referencia deben estar entre comillas dobles para que el compilador YAML las interprete correctamente.

Tipos de constructos en bucle

Esta sección contiene información y ejemplos sobre los tipos de constructos en bucle que se pueden utilizar en TOE de AWS.

Tipos de constructos en bucle

- Bucle for
- Bucle forEach con lista de entradas
- Bucle forEach con lista delimitada

Bucle **`for`**

El bucle `for` itera en un rango de enteros especificado dentro de un límite delimitado por el inicio y el final de las variables. Los valores iterativos están en el conjunto `[start, end]` e incluyen los valores límite.

TOE de AWS verifica los `updateBy` valores `startend`, y para garantizar que la combinación no dé como resultado un bucle infinito.

Esquema de bucle `for`

```
  - name: "StepName"
    action: "ActionModule"
    loop:
      name: "string"
      for:
        start: int
        end: int
        updateBy: int
inputs:
  ...
```

Entrada de bucle **`for`**

Campo	Descripción	Tipo	Obligatorio/a	Predeterminado/a
`name`	Nombre único del bucle. Debe ser único en comparación con otros nombres de bucles de la misma fase.	Cadena	No	""
`start`	Valor inicial de la iteración. No acepta expresiones encadenadas.	Entero	Sí	n/a
`end`	Valor final de la iteración. No acepta expresiones encadenadas.	Entero	Sí	n/a
`updateBy`	Diferencia por la que un valor iterativo se actualiza	Entero	Sí	n/a

Campo	Descripción	Tipo	Obligatorio/a	Predeterminado/a
	mediante la adición. Debe ser un valor negativo o positivo distinto de cero. No acepta expresiones encadenadas.			

Ejemplo de entrada de bucle `for`

```
  - name: "CalculateFileUploadLatencies"
    action: "ExecutePowerShell"
    loop:
      for:
        start: 100000
        end: 1000000
        updateBy: 100000
    inputs:
      commands:
        - |
          $f = new-object System.IO.FileStream c:\temp\test{{ loop.index }}.txt,
 Create, ReadWrite
          $f.SetLength({{ loop.value }}MB)
          $f.Close()
        - c:\users\administrator\downloads\latencyTest.exe --file c:\temp
\test{{ loop.index }}.txt
        - AWS s3 cp c:\users\administrator\downloads\latencyMetrics.json s3://bucket/
latencyMetrics.json
        - |
          Remove-Item -Path c:\temp\test{{ loop.index }}.txt
          Remove-Item -Path c:\users\administrator\downloads\latencyMetrics.json
```

Bucle **`forEach`** con lista de entradas

El bucle `forEach` itera en una lista explícita de valores, que pueden ser cadenas y expresiones encadenadas.

Bucle `forEach` con esquema de lista de entrada

```
- name: "StepName"
  action: "ActionModule"
  loop:
    name: "string"
    forEach:
      - "string"
  inputs:
...
```

Bucle **`forEach`** con entrada de lista de entrada

Campo	Descripción	Tipo	Obligatorio/a	Predeterminado/a
`name`	Nombre único del bucle. Debe ser único en comparación con otros nombres de bucles de la misma fase.	Cadena	No	""
Lista de cadenas de bucle `forEach`	Lista de cadenas para la iteración . Acepta expresiones encadenadas como cadenas en la lista. Las expresiones encadenadas deben estar entre comillas dobles para que el compilador YAML las	Lista de cadenas	Sí	n/a

Campo	Descripción	Tipo	Obligatorio/a	Predeterminado/a
	interprete correctamente.			

Bucle `forEach` con lista de entradas (ejemplo 1)

```
- name: "ExecuteCustomScripts"
  action: "ExecuteBash"
  loop:
    name: BatchExecLoop
    forEach:
      - /tmp/script1.sh
      - /tmp/script2.sh
      - /tmp/script3.sh
  inputs:
    commands:
      - echo "Count {{ BatchExecLoop.index }}"
      - sh "{{ loop.value }}"
      - |
        retVal=$?
        if [ $retVal -ne 0 ]; then
          echo "Failed"
        else
          echo "Passed"
       fi
```

Bucle `forEach` con lista de entradas (ejemplo 2)

```
- name: "RunMSIWithDifferentArgs"
  action: "ExecuteBinary"
  loop:
    name: MultiArgLoop
    forEach:
      - "ARG1=C:\Users ARG2=1"
      - "ARG1=C:\Users"
      - "ARG1=C:\Users ARG3=C:\Users\Administrator\Documents\f1.txt"
  inputs:
    commands:
      path: "c:\users\administrator\downloads\runner.exe"
```

```
      args:
        - "{{ MultiArgLoop.value }}"
```

Bucle `forEach` con lista de entradas (ejemplo 3)

```
- name: "DownloadAllBinaries"
  action: "S3Download"
  loop:
    name: MultiArgLoop
    forEach:
      - "bin1.exe"
      - "bin10.exe"
      - "bin5.exe"
  inputs:
    - source: "s3://bucket/{{ loop.value }}"
      destination: "c:\temp\{{ loop.value }}"
```

Bucle **`forEach`** con lista delimitada

El bucle itera sobre una cadena que contiene valores separados por un delimitador. Para recorrer en iteración los componentes de la cadena, TOE de AWS utiliza el delimitador para dividir la cadena en una matriz adecuada para la iteración.

Bucle `forEach` con esquema de lista delimitada

```
- name: "StepName"
  action: "ActionModule"
  loop:
    name: "string"
    forEach:
      list: "string"
      delimiter: ".,;:\n\t -_"
  inputs:
...
```

Bucle **`forEach`** con entrada de lista delimitada

Campo	Descripción	Tipo	Obligatorio/a	Predeterminado/a
`name`	Nombre único del bucle. Debe	Cadena	No	""

Campo	Descripción	Tipo	Obligatorio/a	Predeterminado/a
	ser único en comparación con otros nombres de bucles de la misma fase.			
`list`	Cadena compuesta por cadenas constitutivas unidas por un carácter delimitador común. También acepta expresiones encadenadas. En el caso de expresiones encadenadas, asegúrese de que estén entre comillas dobles para que el compilador de YAML las interprete correctamente.	Cadena	Sí	n/a

Campo	Descripción	Tipo	Obligatorio/a	Predeterminado/a
`delimiter`	Carácter que se usa para separar las cadenas dentro de un bloque. El valor predeterminado es el carácter de coma. Solo se permite un carácter delimitador de la lista dada: • Punto: `"."` • Coma: `","` • Punto y coma: `";"` • Dos puntos: `":"` • Nueva línea: `"\n"` • Tabulador: `"\t"` • Espacio: `" "` • Guion: `"-"` •	Cadena	No	Coma: `","`

Campo	Descripción	Tipo	Obligatorio/a	Predeterminado/a
	Guion bajo: "_" No se pueden utilizar expresiones encadenadas.			

> **Note**
>
> El valor de `list` se trata como una cadena inmutable. Si la fuente de `list` se cambia durante el tiempo de ejecución, no se reflejará durante la ejecución.

Bucle `forEach` con lista delimitada (ejemplo 1)

Este ejemplo utiliza el siguiente patrón de expresión de encadenamiento para hacer referencia al resultado de otro paso: `<phase_name>.<step_name>.[inputs | outputs].<var_name>`.

```
- name: "RunMSIs"
  action: "ExecuteBinary"
  loop:
    forEach:
      list: "{{ build.GetAllMSIPathsForInstallation.outputs.stdout }}"
      delimiter: "\n"
  inputs:
    commands:
      path: "{{ loop.value }}"
```

Bucle `forEach` con lista delimitada (ejemplo 2)

```
- name: "UploadMetricFiles"
  action: "S3Upload"
  loop:
    forEach:
      list: "/tmp/m1.txt,/tmp/m2.txt,/tmp/m3.txt,..."
```

```
    inputs:
      commands:
        - source: "{{ loop.value }}"
          destination: "s3://bucket/key/{{ loop.value }}"
```

Campos de pasos

Los bucles son parte de un paso. Los campos relacionados con la ejecución de un paso no se aplican a las iteraciones individuales. Los campos de paso se aplican solo a nivel del paso, de la siguiente manera:

- timeoutSeconds: todas las iteraciones del bucle deben ejecutarse dentro del período de tiempo especificado en este campo. Si se agota el tiempo de espera del bucle, TOE de AWS ejecuta la política de reintentos del paso y restablece el parámetro de tiempo de espera para cada nuevo intento. Si la ejecución del bucle supera el valor de tiempo de espera tras alcanzar el número máximo de reintentos, el mensaje de error del paso indica que se ha agotado el tiempo de espera de la ejecución del bucle.
- onFailure: la administración de errores se aplica al paso de la siguiente manera:
  - Si OnFailure está establecido en`Abort`, TOE de AWS sale del bucle y vuelve a intentar el paso de acuerdo con la política de reintentos. Tras el número máximo de reintentos, TOE de AWS marca el paso actual como fallido y detiene la ejecución del proceso.

    TOE de AWS establece el código de estado de la fase principal y el documento en. `Failed`

    > **Note**
    >
    > No se ejecutarán más pasos después del paso con errores.

  - Si onFailure está establecido como `Continue`, TOE de AWS sale del bucle y vuelve a intentar el paso de acuerdo con la política de reintentos. Tras el número máximo de reintentos, TOE de AWS marca el paso actual como fallido y continúa con el paso siguiente.

    TOE de AWS establece el código de estado de la fase principal y el documento en. `Failed`
  - Si onFailure está establecido como `Ignore`, TOE de AWS sale del bucle y vuelve a intentar el paso de acuerdo con la política de reintentos. Tras el número máximo de reintentos, TOE de AWS marca el paso actual como `IgnoredFailure` y continúa hasta ejecutar el paso siguiente.

    TOE de AWS establece el código de estado de la fase principal y el documento en. `SuccessWithIgnoredFailure`

> **Note**
>
> Esto sigue considerándose una ejecución correcta, pero incluye información que le permite saber que uno o más pasos tienen errores y se ignoraron.

- maxAttempts: para cada reintento, todo el paso y todas las iteraciones se ejecutan desde el principio.
- status: el estado general de la ejecución de un paso. `status` no representa el estado de las iteraciones individuales. El estado de un paso con bucles se determina de la siguiente manera:
  - Si no se puede ejecutar una sola iteración, el estado de un paso indica que se ha producido un error.
  - Si todas las iteraciones se realizan correctamente, el estado de un paso indica que se ha realizado correctamente.
- startTime: hora de inicio general de la ejecución de un paso. No representa la hora de inicio de las iteraciones individuales.
- endTime: la hora de finalización general de la ejecución de un paso. No representa la hora de finalización de las iteraciones individuales.
- failureMessage: incluye los índices de iteración que fallaron en caso de errores sin tiempo de espera. En caso de errores de tiempo de espera, el mensaje indica que se produjo un error en la ejecución del bucle. No se proporcionan mensajes de error individuales para cada iteración para minimizar el tamaño de los mensajes de error.

Salidas de paso e iteración

Cada iteración contiene una salida. Al final de una ejecución en bucle, TOE de AWS consolida todos los resultados de la iteración correcta. `detailedOutput.json` Las salidas consolidadas son una recopilación de valores que pertenecen a las claves de salida correspondientes, como se define en el esquema de salida del módulo de acción. El siguiente ejemplo muestra cómo se consolidan las salidas:

Salida de **`ExecuteBash`** para la iteración 1

```
{
 "stdout":"Hello"
}
```

Salida de **ExecuteBash** para la iteración 2

```
{
 "stdout":"World"
}
```

Salida de **ExecuteBash** para el paso

```
{
 "stdout":"Hello\nWorld"
}
```

Por ejemplo, `ExecuteBash`, `ExecutePowerShell` y `ExecuteBinary` son módulos de acción que se devuelven `STDOUT` como salida del módulo de acción. Los mensajes `STDOUT` se unen con el carácter de la nueva línea para producir la salida general del paso en `detailedOutput.json`.

TOE de AWS no consolidará los resultados de las iteraciones fallidas.

Módulos de acción compatibles con el administrador de componentes TOE de AWS

Los servicios de creación de imágenes, como EC2 Image Builder, utilizan módulos de TOE de AWS acción para ayudar a configurar las EC2 instancias que se utilizan para crear y probar imágenes de máquinas personalizadas. En esta sección se describen las características de los módulos de TOE de AWS acción más utilizados y cómo configurarlos, con ejemplos.

los componentes se crean con documentos YAML de texto simple. Para obtener más información sobre sintaxis de documentos, consulte Utilice el marco de documentos de TOE de AWS componentes para componentes personalizados.

> **Note**
>
> Todos los módulos de acción utilizan la misma cuenta que Systems Manager Agent cuando se ejecutan: `root` en Linux y `NT Authority\SYSTEM` en Windows.

La siguiente referencia cruzada clasifica los módulos de acción por el tipo de acciones que realizan.

Ejecución general

- Assert (Linux, Windows, macOS)
- ExecuteBash (Linux, macOS)
- ExecuteBinary (Linux, Windows, macOS)
- ExecuteDocument (Linux, Windows, macOS)
- ExecutePowerShell (Windows)

Módulos de descarga y carga de archivos

- S3Download (Linux, Windows, macOS)
- S3Upload (Linux, Windows, macOS)
- WebDownload (Linux, Windows, macOS)

Operaciones del sistema de archivos

- AppendFile (Linux, Windows, macOS)
- CopyFile (Linux, Windows, macOS)
- CopyFolder (Linux, Windows, macOS)
- CreateFile (Linux, Windows, macOS)
- CreateFolder (Linux, Windows, macOS)
- CreateSymlink (Linux, Windows, macOS)
- DeleteFile (Linux, Windows, macOS)
- DeleteFolder (Linux, Windows, macOS)
- ListFiles (Linux, Windows, macOS)
- MoveFile (Linux, Windows, macOS)
- MoveFolder (Linux, Windows, macOS)
- ReadFile (Linux, Windows, macOS)
- SetFileEncoding (Linux, Windows, macOS)
- SetFileOwner (Linux, Windows, macOS)

- SetFolderOwner (Linux, Windows, macOS)
- SetFilePermissions (Linux, Windows, macOS)
- SetFolderPermissions (Linux, Windows, macOS)

Acciones de instalación de software

- InstallMSI (Windows)
- UninstallMSI (Windows)

Acciones del sistema

- Reboot (Linux, Windows)
- SetRegistry (Windows)
- UpdateOS (Linux, Windows)

Módulos de ejecución general

La siguiente sección contiene detalles sobre los módulos de acción que ejecutan comandos y controlan el flujo de trabajo de ejecución.

Módulos de ejecución general

- Assert (Linux, Windows, macOS)
- ExecuteBash (Linux, macOS)
- ExecuteBinary (Linux, Windows, macOS)
- ExecuteDocument (Linux, Windows, macOS)
- ExecutePowerShell (Windows)

Assert (Linux, Windows, macOS)

El módulo de acción Assert realiza comparaciones de valores utilizando Operadores de comparación o Logical operators (Operadores lógicos) como entrada. El resultado de la expresión del operador (verdadero o falso) indica el estado general de éxito o fracaso del paso.

Si la expresión del operador lógico o de comparación se evalúa como `true`, el paso se marca como `Success`. De lo contrario, el paso se marca como `Failed`. Si el paso falla, el parámetro `onFailure` decide el resultado del paso.

Input

Nombre de la clave	Descripción	Tipo	Obligatorio/a
`input`	Contiene un único operador lógico o de comparación. Tenga en cuenta que los operadores lógicos pueden contener más de un operador de comparación.	Varía en función del operador	Sí

Ejemplo de entrada: una comparación sencilla con el operador de comparación **`stringEquals`**

En este ejemplo se obtiene el resultado `true`.

```
- name: StringComparison
  action: Assert
  inputs:
    stringEquals: '2.1.1'
    value: '{{ validate.ApplicationVersion.outputs.stdout }}'
```

Ejemplo de entrada: comparaciones de expresiones regulares mediante el operador de comparación **`patternMatches`**

Todos estos ejemplos dan como resultado `true`.

```
- name: Letters only
  action: Assert
  inputs:
    patternMatches: '^[a-zA-Z]+$'
    value: 'ThisIsOnlyLetters'

- name: Letters and spaces only
```

```
  action: Assert
  inputs:
    patternMatches: '^[a-zA-Z\s]+$'
    value: 'This text contains spaces'

- name: Numbers only
  action: Assert
  inputs:
    patternMatches: '^[0-9]+$'
    value: '1234567890'
```

Ejemplo de entrada: comparaciones anidadas con operadores lógicos y variables encadenadas

El siguiente ejemplo muestra las comparaciones anidadas con operadores lógicos que utilizan comparaciones con variables encadenadas. `Assert` evalúa `true` si se cumple alguna de las siguientes condiciones:

- La `ApplicationVersion` es mayor que `2.0` y la `CPUArchitecture` igual a `arm64`.
- La `CPUArchitecture` es igual a `x86_64`.

```
- name: NestedComparisons
  action: Assert
  inputs:
    or: # <- first level deep
      - and: # <- second level deep
          - numberGreaterThan: 2.0 # <- third level deep
            value: '{{ validate.ApplicationVersion.outputs.stdout }}'
          - stringEquals: 'arm64'
            value: '{{ validate.CPUArchitecture.outputs.stdout }}'
      - stringEquals: 'x86_64'
        value: '{{ validate.CPUArchitecture.outputs.stdout }}'
```

Salida:

El resultado de un paso `Assert` es el éxito o el fracaso del paso.

ExecuteBash (Linux, macOS)

El módulo de ExecuteBashacción le permite ejecutar scripts bash con códigos/comandos de shell integrados. Este módulo es compatible con Linux.

Todos los comandos e instrucciones que especifique en el bloque de comandos se convierten en un archivo (por ejemplo `input.sh`) y se ejecutan con el intérprete de comandos bash. El resultado de ejecutar el archivo del intérprete de comandos es el código de salida del paso.

El ExecuteBashmódulo gestiona los reinicios del sistema si el script se cierra con un código de salida de. 194 Cuando esto ocurre, la aplicación realiza una de las siguientes acciones:

- La aplicación entrega el código de salida a la persona que llama si lo ejecuta Systems Manager Agent. Systems Manager Agent gestiona el reinicio del sistema y ejecuta el mismo paso que lo inició, tal y como se describe en Reiniciar una instancia gestionada desde scripts.
- La aplicación guarda el `executionstate` actual, configura un activador de reinicio para volver a ejecutar la aplicación y reinicia el sistema.

Tras reiniciar el sistema, la aplicación ejecuta el mismo paso que inició el reinicio. Si necesita esta funcionalidad, debe escribir scripts idempotentes que puedan gestionar múltiples invocaciones del mismo comando del intérprete de comandos.

Input

Nombre de la clave	Descripción	Tipo	Obligatorio/a
`commands`	Contiene una lista de instrucciones o comandos para ejecutar según la sintaxis de bash. Se permite el YAML multilínea.	Enumeración	Sí

Ejemplo de entrada: antes y después de un reinicio

```
name: ExitCode194Example
description: This shows how the exit code can be used to restart a system with
 ExecuteBash
schemaVersion: 1.0
phases:
  - name: build
    steps:
```

```
      - name: RestartTrigger
        action: ExecuteBash
        inputs:
          commands:
            - |
              REBOOT_INDICATOR=/var/tmp/reboot-indicator
              if [ -f "${REBOOT_INDICATOR}" ]; then
                echo 'The reboot file exists. Deleting it and exiting with success.'
                rm "${REBOOT_INDICATOR}"
                exit 0
              fi
              echo 'The reboot file does not exist. Creating it and triggering a
 restart.'
              touch "${REBOOT_INDICATOR}"
              exit 194
```

Output

Campo	Descripción	Tipo
`stdout`	Resultado estándar de la ejecución de comandos.	cadena

Si inicia un reinicio y devuelve el código de salida 194 como parte del módulo de acción, la compilación se reanudará en el mismo paso del módulo de acción en el que se inició el reinicio. Si inicia un reinicio sin el código de salida, es posible que se produzca un error en el proceso de compilación.

Ejemplo de resultado: antes del reinicio (primera vez a través del documento)

```
{
 “stdout”: “The reboot file does not exist. Creating it and triggering a restart."
}
```

Ejemplo de resultado: después del reinicio (segunda vez a través del documento)

```
{
 “stdout”: “The reboot file exists. Deleting it and exiting with success."
}
```

ExecuteBinary (Linux, Windows, macOS)

El módulo de ExecuteBinaryacción permite ejecutar archivos binarios con una lista de argumentos de línea de comandos.

El ExecuteBinarymódulo gestiona los reinicios del sistema si el archivo binario se cierra con un código de salida de `194` (Linux) o `3010` (Windows). Cuando esto ocurre, la aplicación realiza una de las siguientes acciones:

- La aplicación entrega el código de salida a la persona que llama si lo ejecuta Systems Manager Agent. Systems Manager Agent gestiona el reinicio del sistema y ejecuta el mismo paso que lo inició, tal y como se describe en Reiniciar una instancia gestionada desde scripts.
- La aplicación guarda el `executionstate` actual, configura un activador de reinicio para volver a ejecutar la aplicación y reinicia el sistema.

Una vez reiniciado el sistema, la aplicación ejecuta el mismo paso que inició el reinicio. Si necesita esta funcionalidad, debe escribir scripts idempotentes que puedan gestionar múltiples invocaciones del mismo comando del intérprete de comandos.

Input

Nombre de la clave	Descripción	Tipo	Obligatorio/a
`path`	La ruta de acceso al archivo binario que se va a ejecutar.	Cadena	Sí
`arguments`	Contiene una lista de argumentos de la línea de comandos que se utilizarán al ejecutar el archivo binario.	Lista de cadenas	No

Ejemplo de entrada: install.NET

```
- name: "InstallDotnet"
  action: ExecuteBinary
  inputs:
```

```
      path: C:\PathTo\dotnet_installer.exe
      arguments:
        - /qb
        - /norestart
```

Output

Campo	Descripción	Tipo
`stdout`	Resultado estándar de la ejecución de comandos.	cadena

Ejemplo de resultados

```
{
 "stdout": "success"
}
```

ExecuteDocument (Linux, Windows, macOS)

El módulo de ExecuteDocumentacción añade soporte para documentos de componentes anidados, al ejecutar varios documentos de componentes desde un documento. TOE de AWS valida el documento que se pasa en el parámetro de entrada en tiempo de ejecución.

Restricciones

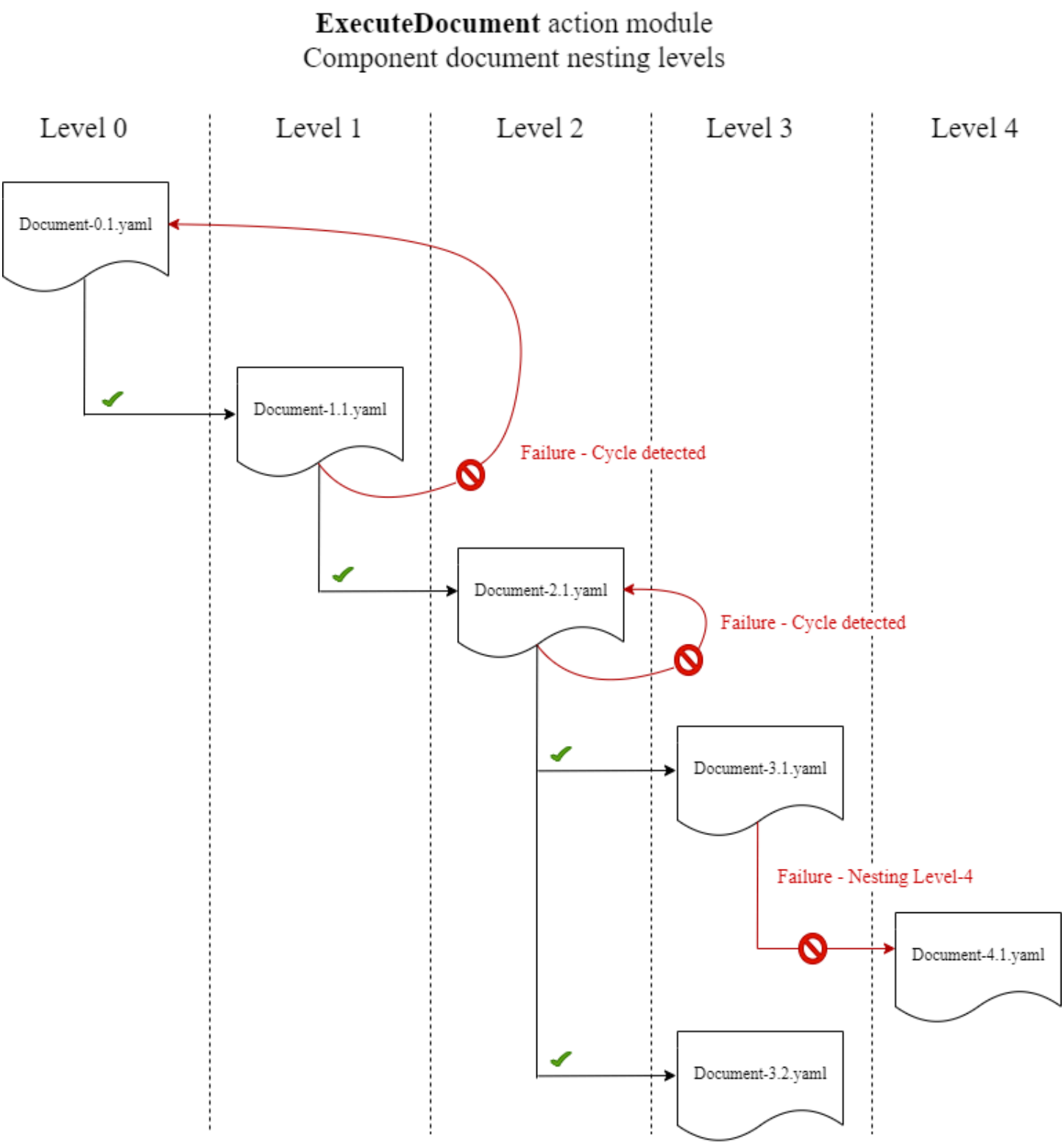
- Este módulo de acción se ejecuta una vez, no se permiten reintentos y no hay opción de establecer límites de tiempo de espera. ExecuteDocumentestablece los siguientes valores predeterminados y devuelve un error si intenta cambiarlos.
  - `timeoutSeconds`: -1
  - `maxAttempts`: 1

  > **Note**
  >
  > Puede dejar estos valores en blanco y TOE de AWS utilizar los valores predeterminados.

- Se permite la anidación de documentos con una profundidad de hasta tres niveles, pero no más. Tres niveles de anidación se traducen en cuatro niveles de documento, ya que el nivel superior

no está anidado. En este escenario, el documento de nivel inferior no debe incluir a ningún otro documento.

- No se permite la ejecución cíclica de los documentos componentes. Cualquier documento que se llame a sí mismo fuera de una construcción en bucle, o que llame a otro documento en un nivel superior de la cadena de ejecución actual, inicia un ciclo que puede dar como resultado un bucle sin fin. Cuando TOE de AWS detecta una ejecución cíclica, la detiene y registra el error.



Si un documento componente intenta ejecutarse por sí mismo o ejecutar alguno de los documentos componentes que se encuentran más arriba en la cadena de ejecución actual, se produce un error en la ejecución.

Entrada

Nombre de la clave	Descripción	Tipo	Obligatorio/a
`document`	Ruta del documento componente. Entre las opciones válidas se incluyen: • Rutas de archivo locales • S3 URIs • EC2 Versión de compilación del componente Image Builder ARNs	Cadena	Sí
`document-s3-bucket-owner`	ID de la cuenta del propietario del bucket de S3 para el bucket de S3 donde se almacenan los documentos de los componentes. (Se recomienda si utiliza S3 URIs en el documento de componentes).	Cadena	No
`phases`	Fases que se ejecutarán en el documento de componentes,	Cadena	No

Nombre de la clave	Descripción	Tipo	Obligatorio/a
	expresadas como una lista separada por comas. Si no se especifica ninguna fase, se ejecutan todas las fases.		
`parameters`	Parámetros de entrada que se pasan al documento del componente en tiempo de ejecución como pares de valores clave.	Lista de mapas de parámetros	No

Entrada de mapa de parámetros

Nombre de la clave	Descripción	Tipo	Obligatorio/a
`name`	El nombre del parámetro de entrada que se va a pasar al documento del componente que está ejecutando el módulo de ExecuteDocumentacción.	Cadena	Sí
`value`	El valor de un parámetro de entrada.	Cadena	Sí

Ejemplos de entradas

Los ejemplos siguientes muestran variaciones de las entradas del documento de componentes, en función de la ruta de instalación.

Ejemplo de entrada: ruta del documento local

```
# main.yaml
schemaVersion: 1.0

phases:
  - name: build
    steps:
      - name: ExecuteNestedDocument
        action: ExecuteDocument
        inputs:
          document: Sample-1.yaml
          phases: build
          parameters:
            - name: parameter-1
              value: value-1
            - name: parameter-2
              value: value-2
```

Ejemplo de entrada: URI de S3 como ruta de documento

```
# main.yaml
schemaVersion: 1.0

phases:
  - name: build
    steps:
      - name: ExecuteNestedDocument
        action: ExecuteDocument
        inputs:
          document: s3://my-bucket/Sample-1.yaml
          document-s3-bucket-owner: 123456789012
          phases: build,validate
          parameters:
            - name: parameter-1
              value: value-1
            - name: parameter-2
              value: value-2
```

Ejemplo de entrada: EC2 ARN del componente Image Builder como ruta de documento

```
# main.yaml
schemaVersion: 1.0

phases:
  - name: build
    steps:
      - name: ExecuteNestedDocument
        action: ExecuteDocument
        inputs:
          document: arn:aws:imagebuilder:us-west-2:aws:component/Sample-Test/1.0.0
          phases: test
          parameters:
            - name: parameter-1
              value: value-1
            - name: parameter-2
              value: value-2
```

Uso de un ForEach bucle para ejecutar documentos

```
# main.yaml
schemaVersion: 1.0

phases:
  - name: build
    steps:
      - name: ExecuteNestedDocument
        action: ExecuteDocument
        loop:
          name: 'myForEachLoop'
          forEach:
            - Sample-1.yaml
            - Sample-2.yaml
        inputs:
          document: "{{myForEachLoop.value}}"
          phases: test
          parameters:
            - name: parameter-1
              value: value-1
            - name: parameter-2
              value: value-2
```

Uso de un bucle For para ejecutar documentos

```
# main.yaml
schemaVersion: 1.0

phases:
  - name: build
    steps:
      - name: ExecuteNestedDocument
        action: ExecuteDocument
        loop:
          name: 'myForLoop'
          for:
            start: 1
            end: 2
            updateBy: 1
        inputs:
          document: "Sample-{{myForLoop.value}}.yaml"
          phases: test
          parameters:
            - name: parameter-1
              value: value-1
            - name: parameter-2
              value: value-2
```

Output

TOE de AWS crea un archivo de salida llamado `detailedoutput.json` cada vez que se ejecuta. El archivo contiene detalles sobre cada fase y paso de cada documento componente que se invoca mientras se está ejecutando. Para el módulo de ExecuteDocumentacciones, encontrará un breve resumen del tiempo de ejecución en el `outputs` campo y detalles sobre las fases, los pasos y los documentos en los que se ejecuta`detailedOutput`.

```
{
 \"executedStepCount\":1,\"executionId\":\"97054e22-06cc-11ec-9b14-acde48001122\",
\"failedStepCount\":0,\"failureMessage\":\"\",\"ignoredFailedStepCount\":0,\"logUrl\":
\"\",\"status\":\"success\"
}",
```

El objeto de resumen de resultados de cada documento componente contiene los siguientes detalles, como se muestra aquí, con valores de ejemplo:

- executedStepCount«:1

- "executionId":"12345a67-89bc-01de-2f34-abcd56789012"
- «failedStepCount«:0
- "failureMessage":""
- «ignoredFailedStepContar»: 0
- "logUrl":""
- "status":"success"

Ejemplo de resultados

El siguiente ejemplo muestra el resultado del módulo de ExecuteDocumentacciones cuando se produce una ejecución anidada. En este ejemplo, el documento componente `main.yaml` ejecuta correctamente el documento componente `Sample-1.yaml`.

```
{
    "executionId": "12345a67-89bc-01de-2f34-abcd56789012",
    "status": "success",
    "startTime": "2021-08-26T17:20:31-07:00",
    "endTime": "2021-08-26T17:20:31-07:00",
    "failureMessage": "",
    "documents": [
        {
            "name": "",
            "filePath": "main.yaml",
            "status": "success",
            "description": "",
            "startTime": "2021-08-26T17:20:31-07:00",
            "endTime": "2021-08-26T17:20:31-07:00",
            "failureMessage": "",
            "phases": [
                {
                    "name": "build",
                    "status": "success",
                    "startTime": "2021-08-26T17:20:31-07:00",
                    "endTime": "2021-08-26T17:20:31-07:00",
                    "failureMessage": "",
                    "steps": [
                        {
                            "name": "ExecuteNestedDocument",
                            "status": "success",
                            "failureMessage": "",
```

```
                            "timeoutSeconds": -1,
                            "onFailure": "Abort",
                            "maxAttempts": 1,
                            "action": "ExecuteDocument",
                            "startTime": "2021-08-26T17:20:31-07:00",
                            "endTime": "2021-08-26T17:20:31-07:00",
                            "inputs": "[{\"document\":\"Sample-1.yaml\",\"document-s3-
bucket-owner\":\"\",\"phases\":\"\",\"parameters\":null}]",
                            "outputs": "[{\"executedStepCount\":1,\"executionId\":
\"98765f43-21ed-09cb-8a76-fedc54321098\",\"failedStepCount\":0,\"failureMessage\":\"\",
\"ignoredFailedStepCount\":0,\"logUrl\":\"\",\"status\":\"success\"}]",
                            "loop": null,
                            "detailedOutput": [
                                {
                                    "executionId": "98765f43-21ed-09cb-8a76-
fedc54321098",
                                    "status": "success",
                                    "startTime": "2021-08-26T17:20:31-07:00",
                                    "endTime": "2021-08-26T17:20:31-07:00",
                                    "failureMessage": "",
                                    "documents": [
                                        {
                                            "name": "",
                                            "filePath": "Sample-1.yaml",
                                            "status": "success",
                                            "description": "",
                                            "startTime": "2021-08-26T17:20:31-07:00",
                                            "endTime": "2021-08-26T17:20:31-07:00",
                                            "failureMessage": "",
                                            "phases": [
                                                {
                                                    "name": "build",
                                                    "status": "success",
                                                    "startTime":
 "2021-08-26T17:20:31-07:00",
                                                    "endTime":
 "2021-08-26T17:20:31-07:00",
                                                    "failureMessage": "",
                                                    "steps": [
                                                        {
                                                            "name": "ExecuteBashStep",
                                                            "status": "success",
                                                            "failureMessage": "",
                                                            "timeoutSeconds": 7200,
```

```
                                                                "onFailure": "Abort",
                                                                "maxAttempts": 1,
                                                                "action": "ExecuteBash",
                                                                "startTime":
 "2021-08-26T17:20:31-07:00",
                                                                "endTime":
 "2021-08-26T17:20:31-07:00",
                                                                "inputs": "[{\"commands\":
[\"echo \\\"Hello World!\\\"\"]}]",
                                                                "outputs": "[{\"stdout\":
\"Hello World!\"}]",
                                                                "loop": null,
                                                                "detailedOutput": null
                                                            }]
                                                        }]
                                                    }]
                                                }]
                                            }]

                                    }]
                                }]
}
```

ExecutePowerShell (Windows)

El módulo de ExecutePowerShellacción permite ejecutar PowerShell scripts con códigos o comandos de shell integrados. Este módulo es compatible con la plataforma Windows y Windows. PowerShell

Todo lo commands/instructions especificado en el bloque de comandos se convierte en un archivo de script (por ejemplo`input.ps1`) y se ejecuta con WindowsPowerShell. El resultado de ejecutar el archivo del intérprete de comandos es el código de salida.

El ExecutePowerShellmódulo gestiona los reinicios del sistema si el comando shell se cierra con un código de salida de. `3010` Cuando esto ocurre, la aplicación realiza una de las siguientes acciones:

- La aplicación entrega el código de salida a la persona que llama si lo ejecuta el Systems Manager Agent. Systems Manager Agent gestiona el reinicio del sistema y ejecuta el mismo paso que lo inició, tal y como se describe en Reiniciar una instancia gestionada desde scripts.
- La aplicación guarda el `executionstate` actual, configura un activador de reinicio para volver a ejecutar la aplicación y reinicia el sistema.

Tras reiniciar el sistema, la aplicación ejecuta el mismo paso que inició el reinicio. Si necesita esta funcionalidad, debe escribir scripts idempotentes que puedan gestionar múltiples invocaciones del mismo comando del intérprete de comandos.

Input

Nombre de la clave	Descripción	Tipo	Obligatorio/a
`commands`	Contiene una lista de instrucciones o comandos que se deben ejecutar según PowerShell la sintaxis. Se permite el YAML multilínea.	Lista de cadenas	Sí. Debe especificar los `commands` o el `file` no ambos.
`file`	Contiene la ruta a un archivo de PowerShel l script. PowerShell se ejecutará en este archivo mediante el argumento de la línea de `-file` comandos. La ruta debe apuntar a un archivo `.ps1`.	Cadena	Sí. Debe especificar los `commands` o el `file` no ambos.

Ejemplo de entrada: antes y después de un reinicio

```
name: ExitCode3010Example
description: This shows how the exit code can be used to restart a system with
 ExecutePowerShell
schemaVersion: 1.0
phases:
  - name: build
    steps:
      - name: RestartTrigger
        action: ExecutePowerShell
        inputs:
          commands:
```

```
            - |
              $rebootIndicator = Join-Path -Path $env:SystemDrive -ChildPath 'reboot-
indicator'
              if (Test-Path -Path $rebootIndicator) {
                Write-Host 'The reboot file exists. Deleting it and exiting with
 success.'
                Remove-Item -Path $rebootIndicator -Force | Out-Null
                [System.Environment]::Exit(0)
              }
              Write-Host 'The reboot file does not exist. Creating it and triggering a
 restart.'
              New-Item -Path $rebootIndicator -ItemType File | Out-Null
              [System.Environment]::Exit(3010)
```

Output

Campo	Descripción	Tipo
`stdout`	Resultado estándar de la ejecución de comandos.	cadena

Si ejecuta un reinicio y devuelve el código de salida `3010` como parte del módulo de acción, la compilación se reanudará en el mismo paso del módulo de acción en el que se inició el reinicio. Si ejecuta un reinicio sin el código de salida, es posible que se produzca un error en el proceso de compilación.

Ejemplo de resultado: antes del reinicio (primera vez a través del documento)

```
{
 “stdout”: “The reboot file does not exist. Creating it and triggering a restart."
}
```

Ejemplo de resultado: después del reinicio (segunda vez a través del documento)

```
{
 “stdout”: “The reboot file exists. Deleting it and exiting with success."
}
```

Módulos de descarga y carga de archivos

La siguiente sección contiene detalles sobre los módulos de acción que cargan o descargan archivos.

Descarga y carga los módulos de acción

- S3Download (Linux, Windows, macOS)
- S3Upload (Linux, Windows, macOS)
- WebDownload (Linux, Windows, macOS)

S3Download (Linux, Windows, macOS)

Con el módulo de acción `S3Download`, puede descargar un objeto de Amazon S3, o un conjunto de objetos, a un archivo o carpeta local que especifique con la ruta `destination`. Si ya existe algún archivo en la ubicación especificada y el indicador `overwrite` está establecido como true, `S3Download` sobrescribe el archivo.

Su ubicación `source` puede apuntar a un objeto específico en Amazon S3, o puede usar un prefijo de clave con un asterisco como comodín (`*`) para descargar un conjunto de objetos que coincidan con la ruta del prefijo de clave. Al especificar un prefijo de clave en su ubicación `source`, el módulo de acción `S3Download` descarga todo lo que coincida con el prefijo (archivos y carpetas incluidos). Asegúrese de que el prefijo de clave termine con una barra diagonal seguida de un asterisco (`/*`), para descargar todo lo que coincida con el prefijo. Por ejemplo: *`s3://my-bucket/my-folder/*`*.

Si la acción `S3Download` de un prefijo de clave especificado falla durante una descarga, el contenido de la carpeta no vuelve a su estado anterior al error. La carpeta de destino permanece tal y como estaba en el momento del error.

Casos de uso admitidos

El módulo de acción `S3Download` admite los siguientes casos de uso:

- El objeto Amazon S3 se descarga en una carpeta local, tal y como se especifica en la ruta de descarga.
- Los objetos de Amazon S3 (con un prefijo de clave en la ruta del archivo de Amazon S3) se descargan en la carpeta local especificada, que copia de forma recursiva todos los objetos de Amazon S3 que coincidan con el prefijo de la clave en la carpeta local.

Requisitos de IAM

El rol de IAM que asocie al perfil de instancia debe tener permiso para ejecutar el módulo de acción `S3Download`. Las siguientes políticas de IAM se deben adjuntar al rol de IAM asociado al perfil de instancia:

- Archivo único: `s3:GetObject` contra bucket/object (por ejemplo,`arn:aws:s3:::BucketName/*`).
- Varios archivos: `s3:ListBucket` contra bucket/object (por ejemplo,`arn:aws:s3:::BucketName`) y `s3:GetObject` contra bucket/object (por ejemplo,`arn:aws:s3:::BucketName/*`).

Input

Clave	Descripción	Tipo	Obligatorio/a	Predeterminado/a
`source`	El bucket de Amazon S3 que es el origen de la descarga. Puede especificar una ruta a un objeto específico o utilizar un prefijo de clave que termine con una barra diagonal seguida de un asterisco comodín (`/*`), para descargar un conjunto de objetos que coincidan con el prefijo de clave.	Cadena	Sí	N/A

Clave	Descripción	Tipo	Obligatorio/a	Predeterminado/a
`destination`	La ruta local en la que se descargan los objetos de Amazon S3. Para descargar un solo archivo, debe especificarse el nombre del archivo como parte de la ruta. Por ejemplo, */myfolder/package.zip* .	Cadena	Sí	N/A
`expectedBucketOwner`	ID de la cuenta de propietario esperada del bucket proporcionado en la ruta `source`. Le recomendamos que compruebe la propiedad del bucket de Amazon S3 especificado en la fuente.	Cadena	No	N/A

Clave	Descripción	Tipo	Obligatorio/a	Predeterminado/a
`overwrite`	Si se establece como true, si ya existe un archivo con el mismo nombre en la carpeta de destino de la ruta local especific ada, el archivo de descarga sobrescribe el archivo local. Si se establece como false, el archivo existente en el sistema local está protegido para que no se sobrescriba y el módulo de acción falla y se produce un error de descarga. Por ejemplo, `Error: S3Download: File already exists and "overwrit e"`	Booleano	No	true

Clave	Descripción	Tipo	Obligatorio/a	Predeterminado/a
	property for "destination" file is set to false. Cannot download.			

> **Note**
>
> En los ejemplos siguientes, la ruta de la carpeta de Windows se puede reemplazar por una ruta de Linux. Por ejemplo, se puede sustituir *C:\myfolder\package.zip* por */myfolder/package.zip*.

Ejemplo de entrada: copia de un objeto de Amazon S3 a un archivo local

En el siguiente ejemplo, se muestra cómo copiar un objeto de Amazon S3 en un archivo local.

```
- name: DownloadMyFile
  action: S3Download
  inputs:
    - source: s3://amzn-s3-demo-source-bucket/path/to/package.zip
      destination: C:\myfolder\package.zip
      expectedBucketOwner: 123456789022
      overwrite: false
    - source: s3://amzn-s3-demo-source-bucket/path/to/package.zip
      destination: C:\myfolder\package.zip
      expectedBucketOwner: 123456789022
      overwrite: true
    - source: s3://amzn-s3-demo-source-bucket/path/to/package.zip
      destination: C:\myfolder\package.zip
      expectedBucketOwner: 123456789022
```

Ejemplo de entrada: copia de todos los objetos de Amazon S3 de un bucket de Amazon S3 con el prefijo de clave a una carpeta local

Ejemplo de entrada: copia todos los objetos de Amazon S3 de un bucket de Amazon S3 con el prefijo de clave a una carpeta local. Amazon S3 no tiene el concepto de carpeta, por lo que se copian todos los objetos que coincidan con el prefijo de clave. El número máximo de objetos que se puede descargar es 1000.

```
- name: MyS3DownloadKeyprefix
  action: S3Download
  maxAttempts: 3
  inputs:
    - source: s3://amzn-s3-demo-source-bucket/path/to/*
      destination: C:\myfolder\
      expectedBucketOwner: 123456789022
      overwrite: false
    - source: s3://amzn-s3-demo-source-bucket/path/to/*
      destination: C:\myfolder\
      expectedBucketOwner: 123456789022
      overwrite: true
    - source: s3://amzn-s3-demo-source-bucket/path/to/*
      destination: C:\myfolder\
      expectedBucketOwner: 123456789022
```

Output

Ninguna.

S3Upload (Linux, Windows, macOS)

Con el módulo de acción S3Upload, puede cargar un archivo desde un archivo o carpeta de origen a una ubicación de Amazon S3. Puede usar un comodín (*) en la ruta especificada para su ubicación de origen para cargar todos los archivos cuya ruta coincida con el patrón de comodín.

Si se produce un error en la acción recursiva S3Upload, los archivos que ya se hayan cargado permanecerán en el bucket de Amazon S3 de destino.

Casos de uso admitidos

- Archivo local del objeto de Amazon S3.
- Archivos locales en la carpeta (con comodín) con el prefijo de clave de Amazon S3.

- Copie la carpeta local (`recurse` debe estar configurado como `true`) al prefijo de clave de Amazon S3.

Requisitos de IAM

El rol de IAM que asocie al perfil de instancia debe tener permiso para ejecutar el módulo de acción `S3Upload`. La siguiente política de IAM debe estar asociada al rol de IAM que está asociado con el perfil de instancia. La política debe conceder permisos `s3:PutObject` al bucket de Amazon S3 de destino. Por ejemplo, `arn:aws:s3:::BucketName/*`).

Input

Clave	Descripción	Tipo	Obligatorio/a	Predeterminado/a
`source`	La ruta local donde se files/folders origina la fuente. `source` admite un asterisco como comodín (`*`).	Cadena	Sí	N/A
`destination`	La ruta del bucket de Amazon S3 de destino en el que se cargan las carpetas o archivos de origen.	Cadena	Sí	N/A
`recurse`	Cuando se establece como `true`, ejecuta	Cadena	No	`false`

Clave	Descripción	Tipo	Obligatorio/a	Predeterminado/a
	S3Upload de forma recursiva.			
`expectedBucketOwner`	El ID de cuenta del propietario esperado para el bucket de Amazon S3 especificado en la ruta de destino. Le recomendamos que compruebe la propiedad del bucket de Amazon S3 especificado en el destino.	Cadena	No	N/A

Ejemplo de entrada: copia de un objeto de Amazon S3 a un archivo local

En el siguiente ejemplo, se muestra cómo copiar un objeto de Amazon S3 en un archivo local.

```
- name: MyS3UploadFile
  action: S3Upload
  onFailure: Abort
  maxAttempts: 3
  inputs:
    - source: C:\myfolder\package.zip
      destination: s3://amzn-s3-demo-destination-bucket/path/to/package.zip
      expectedBucketOwner: 123456789022
```

Ejemplo de entrada: copia de todos los objetos de Amazon S3 de un bucket de Amazon S3 con el prefijo de clave a una carpeta local

Ejemplo de entrada: copia todos los objetos de Amazon S3 de un bucket de Amazon S3 con el prefijo de clave a una carpeta local. En este ejemplo no se copian las subcarpetas ni su contenido porque el `recurse` no está especificado y su valor predeterminado es `false`.

```
- name: MyS3UploadMultipleFiles
  action: S3Upload
  onFailure: Abort
  maxAttempts: 3
  inputs:
    - source: C:\myfolder\*
      destination: s3://amzn-s3-demo-destination-bucket/path/to/
      expectedBucketOwner: 123456789022
```

Ejemplo de entrada: copiar de forma recursiva todos los archivos y carpetas de una carpeta local en un bucket de Amazon S3

El siguiente ejemplo muestra como copiar todos los archivos de forma recursiva de una carpeta local a un bucket de Amazon S3 con el prefijo de clave.

```
- name: MyS3UploadFolder
  action: S3Upload
  onFailure: Abort
  maxAttempts: 3
  inputs:
    - source: C:\myfolder\*
      destination: s3://amzn-s3-demo-destination-bucket/path/to/
      recurse: true
      expectedBucketOwner: 123456789022
```

Output

Ninguna.

WebDownload (Linux, Windows, macOS)

El módulo de WebDownloadacción le permite descargar archivos y recursos desde una ubicación remota a través del HTTP/HTTPS protocolo (se recomienda HTTPS). No hay límites en cuanto al número ni al tamaño de las descargas. Este módulo gestiona la lógica de reintento y retroceso exponencial.

A cada operación de descarga se le asigna un máximo de 5 intentos para que se realice correctamente según las entradas del usuario. Estos intentos difieren de los especificados en el

campo `maxAttempts` del documento `steps` y están relacionados con errores en los módulos de acción.

Este módulo de acción gestiona los redireccionamientos de forma implícita. Todos los códigos de estado HTTP, excepto por `200`, generan un error.

Input

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Predeterminado/a
`source`	La HTTP/HTTPS URL válida (se recomienda HTTPS), que sigue el estándar RFC 3986. Se permiten expresiones de encadenamiento.	Cadena	Sí	N/A
`destination`	Ruta absoluta o relativa de archivos o carpetas del sistema local. Las rutas de las carpetas deben terminar en `/`. Si no terminan en `/`, se tratarán como rutas de archivos. El módulo crea cualquier archivo o carpeta necesarios para que las	Cadena	Sí	N/A

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Predeterminado/a
	descargas se realicen correctamente. Se permiten expresiones de encadenamiento.			

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Predeterminado/a
`overwrite`	Cuando está activado, sobrescribe cualquier archivo existente en el sistema local con el archivo o recurso descargado. Si no está activado, los archivos existentes en el sistema local no se sobrescriben y el módulo de acción produce un error. Cuando la sobreescritura está habilitada y se especific an la suma de verificación y el algoritmo , el módulo de acción de descarga el archivo solo si la suma de verificación y la almohadilla de los archivos preexistentes no coinciden.	Booleano	No	`true`

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Predeterminado/a
`checksum`	Al especificar la suma de comprobación, se compara con la almohadilla del archivo descargado que se genera con el algoritmo suministrado. Para habilitar la verificación de archivos, se deben proporcionar tanto la suma de verificación como el algoritmo. Se permiten expresiones de encadenamiento.	Cadena	No	N/A

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Predeterminado/a
`algorithm`	El algoritmo utilizado para calcular la suma de control. Las opciones sonMD5, SHA1 SHA256, y. SHA512 Para habilitar la verificación de archivos, se deben proporcionar tanto la suma de verificac ión como el algoritmo. Se permiten expresiones de encadenamiento.	Cadena	No	N/A
`ignoreCertificateErrors`	La validación del certificado SSL se ignora cuando está habilitada.	Booleano	No	`false`

Output

Nombre de la clave	Descripción	Tipo				
`destination`	Cadena de nueva línea	Cadena				

Nombre de la clave	Descripción	Tipo				
	delimitada por caracteres que especifica la ruta de destino en la que se almacenan los archivos o recursos descargados.					

Ejemplo de entrada: descarga del archivo remoto al destino local

```
  - name: DownloadRemoteFile
    action: WebDownload
    maxAttempts: 3
    inputs:
      - source: https://testdomain/path/to/java14.zip
        destination: C:\testfolder\package.zip
```

Salida:

```
{
 "destination": "C:\\testfolder\\package.zip"
}
```

Ejemplo de entrada: descarga de más de un archivo remoto a más de un destino local

```
  - name: DownloadRemoteFiles
    action: WebDownload
    maxAttempts: 3
```

```
    inputs:
      - source: https://testdomain/path/to/java14.zip
        destination: /tmp/java14_renamed.zip
      - source: https://testdomain/path/to/java14.zip
        destination: /tmp/create_new_folder_and_add_java14_as_zip/
```

Salida:

```
{
 "destination": "/tmp/create_new_folder/java14_renamed.zip\n/tmp/
create_new_folder_and_add_java14_as_zip/java14.zip"
}
```

Ejemplo de entrada: descarga de un archivo remoto sin sobrescribir el destino local y descargar otro archivo remoto con verificación de archivos

```
  - name: DownloadRemoteMultipleProperties
    action: WebDownload
    maxAttempts: 3
    inputs:
      - source: https://testdomain/path/to/java14.zip
        destination: C:\create_new_folder\java14_renamed.zip
        overwrite: false
      - source: https://testdomain/path/to/java14.zip
        destination: C:\create_new_folder_and_add_java14_as_zip\
        checksum: ac68bbf921d953d1cfab916cb6120864
        algorithm: MD5
        overwrite: true
```

Salida:

```
{
 "destination": "C:\\create_new_folder\\java14_renamed.zip\nC:\
\create_new_folder_and_add_java14_as_zip\\java14.zip"
}
```

Ejemplo de entrada: descarga de un archivo remoto e ignore la validación de la certificación SSL

```
  - name: DownloadRemoteIgnoreValidation
    action: WebDownload
    maxAttempts: 3
```

```
    inputs:
      - source: https://www.bad-ssl.com/resource
        destination: /tmp/downloads/
        ignoreCertificateErrors: true
```

Salida:

```
{
 "destination": "/tmp/downloads/resource"
}
```

Módulos de operaciones del sistema de archivos

La siguiente sección contiene detalles sobre los módulos de acción que realizan operaciones en el sistema de archivos.

Módulos de acción de operación del sistema de archivos

- AppendFile (Linux, Windows, macOS)
- CopyFile (Linux, Windows, macOS)
- CopyFolder (Linux, Windows, macOS)
- CreateFile (Linux, Windows, macOS)
- CreateFolder (Linux, Windows, macOS)
- CreateSymlink (Linux, Windows, macOS)
- DeleteFile (Linux, Windows, macOS)
- DeleteFolder (Linux, Windows, macOS)
- ListFiles (Linux, Windows, macOS)
- MoveFile (Linux, Windows, macOS)
- MoveFolder (Linux, Windows, macOS)
- ReadFile (Linux, Windows, macOS)
- SetFileEncoding (Linux, Windows, macOS)
- SetFileOwner (Linux, Windows, macOS)
- SetFolderOwner (Linux, Windows, macOS)
- SetFilePermissions (Linux, Windows, macOS)
- SetFolderPermissions (Linux, Windows, macOS)

AppendFile (Linux, Windows, macOS)

El módulo de AppendFileacción añade contenido específico al contenido preexistente de un archivo.

Si el valor de codificación del archivo es diferente del valor de codificación (`utf-8`) predeterminado, puede especificar el valor de codificación del archivo mediante la opción `encoding`. De forma predeterminada, se supone que `utf-16` y `utf-32` utilizan la codificación little-endian.

El módulo de acción devuelve un error cuando ocurre lo siguiente:

- El archivo especificado no existe en tiempo de ejecución.
- No tiene permisos de escritura para modificar el contenido del archivo.
- El módulo detecta un error durante la operación del archivo.

Input

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables	Compatible con todas las plataformas
`path`	La ruta del archivo.	Cadena	Sí	N/A	N/A	Sí
`content`	El contenido que se va a adjuntar al archivo.	Cadena	No	Cadena vacía	N/A	Sí
`encoding`	El estándar de la codificación.	Cadena	No	`utf8`	`utf8`, `utf-8`, `utf16`,`utf-16-LE`, `utf-16-LE` `utf16-BE`,	Sí

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables	Compatible con todas las plataformas
					`utf-16-BE` , `utf32`, `utf-32`, `utf32-LE`,`utf-32-LE` , `utf32-BE`, y `utf-32-BE` . El valor de la opción de codificación no distingue entre mayúsculas y minúsculas.	

Ejemplo de entrada: añadir un archivo sin codificar (Linux)

```
- name: AppendingFileWithOutEncodingLinux
  action: AppendFile
  inputs:
    - path: ./Sample.txt
      content: "The string to be appended to the file"
```

Ejemplo de entrada: añadir un archivo sin codificar (Windows)

```
- name: AppendingFileWithOutEncodingWindows
  action: AppendFile
  inputs:
    - path: C:\MyFolder\MyFile.txt
      content: "The string to be appended to the file"
```

Ejemplo de entrada: añadir un archivo con codificación (Linux)

```
- name: AppendingFileWithEncodingLinux
  action: AppendFile
  inputs:
    - path: /FolderName/SampleFile.txt
      content: "The string to be appended to the file"
      encoding: UTF-32
```

Ejemplo de entrada: añadir un archivo con codificación (Windows)

```
- name: AppendingFileWithEncodingWindows
  action: AppendFile
  inputs:
    - path: C:\MyFolderName\SampleFile.txt
      content: "The string to be appended to the file"
      encoding: UTF-32
```

Ejemplo de entrada: añadir un archivo con una cadena vacía (Linux)

```
- name: AppendingEmptyStringLinux
  action: AppendFile
  inputs:
    - path: /FolderName/SampleFile.txt
```

Ejemplo de entrada: añadir un archivo con una cadena vacía (Windows)

```
- name: AppendingEmptyStringWindows
  action: AppendFile
  inputs:
    - path: C:\MyFolderName\SampleFile.txt
```

Output

Ninguna.

CopyFile (Linux, Windows, macOS)

El módulo de CopyFileacción copia los archivos de la fuente especificada al destino especificado. De forma predeterminada, el módulo crea de forma recursiva la carpeta de destino si no existe en tiempo de ejecución.

Si ya existe un archivo con el nombre especificado en la carpeta especificada, el módulo de acción, de forma predeterminada, sobrescribe el archivo existente. Puede anular este comportamiento al configurar la opción sobrescribir como `false`. Cuando la opción de sobreescritura esté establecida como `false` y ya haya un archivo en la ubicación especificada con el nombre especificado, el módulo de acción devolverá un error. Esta opción funciona igual que el comando de Linux `cp`, que se sobrescribe de forma predeterminada.

El nombre del archivo fuente puede incluir un comodín (`*`). Los caracteres comodín solo se aceptan después del último separador de ruta del archivo (`/` o `\`). Si se incluyen caracteres comodín en el nombre del archivo de origen, todos los archivos que coincidan con el comodín se copian en la carpeta de destino. Si desea mover más de un archivo con un carácter comodín, la entrada de la opción `destination` debe terminar con un separador de rutas de archivo (`/` o `\`), lo que indica que la entrada de destino es una carpeta.

Si el nombre del archivo de destino es diferente del nombre del archivo de origen, puede especificar el nombre del archivo de destino mediante la opción `destination`. Si no especifica un nombre de archivo de destino, se utilizará el nombre del archivo de origen para crear el archivo de destino. Cualquier texto que siga al separador (`/` o `\`) de la última ruta del archivo se trata como el nombre del archivo. Si desea utilizar el mismo nombre de archivo que el archivo de origen, la entrada de la opción `destination` debe terminar con un separador de rutas de archivo (`/` o `\`).

El módulo de acción devuelve un error cuando ocurre lo siguiente:

- No tiene permiso para crear un archivo en la carpeta especificada.
- Los archivos de origen no existen en tiempo de ejecución.
- Ya existe una carpeta con el nombre de archivo especificado y la opción `overwrite` está configurada como `false`.
- El módulo de acción detecta un error al realizar la operación.

Input

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables	Compatible con todas las plataformas
`source`	La ruta del archivo de origen.	Cadena	Sí	N/A	N/A	Sí
`destination`	La ruta del archivo de destino.	Cadena	Sí	N/A	N/A	Sí
`overwrite`	Si se establece como falso, los archivos de destino no se reemplazarán cuando ya haya un archivo en la ubicación especificada con el nombre especificado.	Booleano	No	`true`	N/A	Sí

Ejemplo de entrada: copiar un archivo (Linux)

```
- name: CopyingAFileLinux
  action: CopyFile
  inputs:
    - source: /Sample/MyFolder/Sample.txt
      destination: /MyFolder/destinationFile.txt
```

Ejemplo de entrada: copiar un archivo (Windows)

```
- name: CopyingAFileWindows
  action: CopyFile
  inputs:
    - source: C:\MyFolder\Sample.txt
      destination: C:\MyFolder\destinationFile.txt
```

Ejemplo de entrada: copia un archivo con el nombre del archivo fuente (Linux)

```
- name: CopyingFileWithSourceFileNameLinux
  action: CopyFile
  inputs:
    - source: /Sample/MyFolder/Sample.txt
      destination: /MyFolder/
```

Ejemplo de entrada: copia un archivo con el nombre del archivo fuente (Windows)

```
- name: CopyingFileWithSourceFileNameWindows
  action: CopyFile
  inputs:
    - source: C:\Sample\MyFolder\Sample.txt
      destination: C:\MyFolder\
```

Ejemplo de entrada: copia un archivo con el carácter comodín (Linux)

```
- name: CopyingFilesWithWildCardLinux
  action: CopyFile
  inputs:
    - source: /Sample/MyFolder/Sample*
      destination: /MyFolder/
```

Ejemplo de entrada: copia un archivo con el carácter comodín (Windows)

```
- name: CopyingFilesWithWildCardWindows
```

```
    action: CopyFile
    inputs:
      - source: C:\Sample\MyFolder\Sample*
        destination: C:\MyFolder\
```

Ejemplo de entrada: copiar un archivo sin sobrescribirlo (Linux)

```
  - name: CopyingFilesWithoutOverwriteLinux
    action: CopyFile
    inputs:
      - source: /Sample/MyFolder/Sample.txt
        destination: /MyFolder/destinationFile.txt
        overwrite: false
```

Ejemplo de entrada: copiar un archivo sin sobrescribirlo (Windows)

```
  - name: CopyingFilesWithoutOverwriteWindows
    action: CopyFile
    inputs:
      - source: C:\Sample\MyFolder\Sample.txt
        destination: C:\MyFolder\destinationFile.txt
        overwrite: false
```

Output

Ninguna.

CopyFolder (Linux, Windows, macOS)

El módulo de CopyFolderacción copia una carpeta de la fuente especificada al destino especificado. La entrada de la opción `source` es la carpeta que se va a copiar y la entrada de la opción `destination` es la carpeta en la que se copia el contenido de la carpeta de origen. De forma predeterminada, el módulo crea de forma recursiva la carpeta de destino si no existe en tiempo de ejecución.

Si ya existe una carpeta con el nombre especificado en la carpeta especificada, el módulo de acción, de forma predeterminada, sobrescribe la carpeta existente. Puede anular este comportamiento al configurar la opción sobrescribir como `false`. Cuando la opción de sobreescritura esté establecida como `false` y ya haya una carpeta en la ubicación especificada con el nombre especificado, el módulo de acción devolverá un error.

El nombre de la carpeta fuente puede incluir un comodín (*). Los caracteres comodín solo se aceptan después del último separador de ruta del archivo (/ o \). Si se incluyen caracteres comodín en el nombre de la carpeta de origen, todas las carpetas que coincidan con el comodín se copian en la carpeta de destino. Si desea mover más de una carpeta utilizando un carácter comodín, la entrada de la opción `destination` debe terminar con un separador de rutas de archivo (/ o \), lo que indica que la entrada de destino es una carpeta.

Si el nombre de la carpeta de destino es diferente del nombre de la carpeta de origen, puede especificar el nombre de la carpeta de destino mediante la opción `destination`. Si no especifica un nombre de carpeta de destino, se utilizará el nombre de la carpeta de origen para crear la carpeta de destino. Cualquier texto que siga al separador (/ o \) de la última ruta del archivo se trata como el nombre de la carpeta. Si desea utilizar el mismo nombre de carpeta que la carpeta de origen, la entrada de la opción `destination` debe terminar con un separador de rutas de archivo (/ o \).

El módulo de acción devuelve un error cuando ocurre lo siguiente:

- No tiene permiso para crear una carpeta en la carpeta especificada.
- Las carpetas de origen no existen en tiempo de ejecución.
- Ya existe una carpeta con el nombre de carpeta especificado y la opción `overwrite` está configurada como `false`.
- El módulo de acción detecta un error al realizar la operación.

Input

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables	Compatible con todas las plataformas
`source`	La ruta de la carpeta de origen.	Cadena	Sí	N/A	N/A	Sí
`destination`	La ruta de la carpeta de destino.	Cadena	Sí	N/A	N/A	Sí

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables	Compatible con todas las plataformas
`overwrite`	Si se establece como falso, las carpetas de destino no se reemplazarán cuando ya haya una carpeta en la ubicación especificada con el nombre especificado.	Booleano	No	`true`	N/A	Sí

Ejemplo de entrada: copiar una carpeta (Linux)

```
- name: CopyingAFolderLinux
  action: CopyFolder
  inputs:
    - source: /Sample/MyFolder/SampleFolder
      destination: /MyFolder/destinationFolder
```

Ejemplo de entrada: copiar una carpeta (Windows)

```
- name: CopyingAFolderWindows
```

```
  action: CopyFolder
  inputs:
    - source: C:\Sample\MyFolder\SampleFolder
      destination: C:\MyFolder\destinationFolder
```

Ejemplo de entrada: copiar una carpeta con el nombre de la carpeta de origen (Linux)

```
- name: CopyingFolderSourceFolderNameLinux
  action: CopyFolder
  inputs:
    - source: /Sample/MyFolder/SourceFolder
      destination: /MyFolder/
```

Ejemplo de entrada: copiar una carpeta con el nombre de la carpeta de origen (Windows)

```
- name: CopyingFolderSourceFolderNameWindows
  action: CopyFolder
  inputs:
    - source: C:\Sample\MyFolder\SampleFolder
      destination: C:\MyFolder\
```

Ejemplo de entrada: copiar una carpeta con el carácter comodín (Linux)

```
- name: CopyingFoldersWithWildCardLinux
  action: CopyFolder
  inputs:
    - source: /Sample/MyFolder/Sample*
      destination: /MyFolder/
```

Ejemplo de entrada: copiar una carpeta con el carácter comodín (Windows)

```
- name: CopyingFoldersWithWildCardWindows
  action: CopyFolder
  inputs:
    - source: C:\Sample\MyFolder\Sample*
      destination: C:\MyFolder\
```

Ejemplo de entrada: copiar una carpeta sin sobrescribirla (Linux)

```
- name: CopyingFoldersWithoutOverwriteLinux
```

```
    action: CopyFolder
    inputs:
      - source: /Sample/MyFolder/SourceFolder
        destination: /MyFolder/destinationFolder
        overwrite: false
```

Ejemplo de entrada: copiar una carpeta sin sobrescribirla (Windows)

```
  - name: CopyingFoldersWithoutOverwrite
    action: CopyFolder
    inputs:
      - source: C:\Sample\MyFolder\SourceFolder
        destination: C:\MyFolder\destinationFolder
        overwrite: false
```

Output

Ninguna.

CreateFile (Linux, Windows, macOS)

El módulo de CreateFileacción crea un archivo en una ubicación específica. De forma predeterminada, si es necesario, el módulo también crea de forma recursiva las carpetas principales.

Si el archivo ya existe en la carpeta especificada, el módulo de acción, de forma predeterminada, trunca o sobrescribe el archivo existente. Puede anular este comportamiento al configurar la opción sobrescribir como `false`. Cuando la opción de sobreescritura esté establecida como `false` y ya haya un archivo en la ubicación especificada con el nombre especificado, el módulo de acción devolverá un error.

Si el valor de codificación del archivo es diferente del valor de codificación (`utf-8`) predeterminado, puede especificar el valor de codificación del archivo mediante la opción `encoding`. De forma predeterminada, se supone que `utf-16` y `utf-32` utilizan la codificación little-endian.

`owner`, `group`, y `permissions` son entradas opcionales. La entrada para `permissions` debe ser un valor de cadena. Los archivos se crean con los valores predeterminados cuando no se proporcionan. Estas opciones no son compatibles con las plataformas Windows. Este módulo de acción valida y devuelve un error si `owner`, `group`, y `permissions` se utilizan en las plataformas Windows.

Este módulo de acción puede crear un archivo con los permisos definidos por el valor `umask` predeterminado del sistema operativo. Debe establecer el valor `umask` si quiere anular el valor predeterminado.

El módulo de acción devuelve un error cuando ocurre lo siguiente:

- No tiene permiso para crear un archivo o una carpeta en la carpeta especificada.
- El módulo de acción detecta un error al realizar la operación.

Input

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables	Compatible con todas las plataformas
`path`	La ruta del archivo.	Cadena	Sí	N/A	N/A	Sí
`content`	El contenido de texto del archivo.	Cadena	No	N/A	N/A	Sí
`encoding`	El estándar de la codificación.	Cadena	No	`utf8`	`utf8`, `utf-8`, `utf16`,`utf-16`, `utf16-LE`, `utf-16-LE` `utf16-BE`, `utf-16-BE` , `utf32`, `utf-32`, `utf32-LE`,`utf-32-`	Sí

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables	Compatible con todas las plataformas
					`LE`, `utf32-BE`, y `utf-32-BE`. El valor de la opción de codificación no distingue entre mayúsculas y minúsculas.	
`owner`	El nombre o el ID de usuario.	Cadena	No	N/A	N/A	No es compatible con Windows.
`group`	El nombre o el ID del grupo.	Cadena	No	El usuario actual.	N/A	No es compatible con Windows.
`permissions`	Permisos de archivos.	Cadena	No	`0666`	N/A	No es compatible con Windows.

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables	Compatible con todas las plataformas
`overwrite`	Si el nombre del archivo especificado ya existe, si se establece este valor como `false` se impide que el archivo se trunque o sobrescriba de forma predeterminada.	Booleano	No	`true`	N/A	Sí

Ejemplo de entrada: crear un archivo sin sobrescribirlo (Linux)

```
- name: CreatingFileWithoutOverwriteLinux
  action: CreateFile
  inputs:
    - path: /home/UserName/Sample.txt
      content: The text content of the sample file.
      overwrite: false
```

Ejemplo de entrada: crear un archivo sin sobrescribirlo (Windows)

```
- name: CreatingFileWithoutOverwriteWindows
  action: CreateFile
  inputs:
    - path: C:\Temp\Sample.txt
      content: The text content of the sample file.
      overwrite: false
```

Ejemplo de entrada: creación de un archivo con las propiedades del archivo

```
- name: CreatingFileWithFileProperties
  action: CreateFile
  inputs:
    - path: SampleFolder/Sample.txt
      content: The text content of the sample file.
      encoding: UTF-16
      owner: Ubuntu
      group: UbuntuGroup
      permissions: 0777
   - path: SampleFolder/SampleFile.txt
      permissions: 755
    - path: SampleFolder/TextFile.txt
      encoding: UTF-16
      owner: root
      group: rootUserGroup
```

Ejemplo de entrada: creación de un archivo sin las propiedades de archivo

```
- name: CreatingFileWithoutFileProperties
  action: CreateFile
  inputs:
    - path: ./Sample.txt
    - path: Sample1.txt
```

Ejemplo de entrada: creación de un archivo vacío para omitir una sección del script de limpieza de Linux

```
- name: CreateSkipCleanupfile
  action: CreateFile
  inputs:
    - path: <skip section file name>
```

Para obtener más información, consulte Anule el script de limpieza de Linux..

Output

Ninguna.

CreateFolder (Linux, Windows, macOS)

El módulo de CreateFolderacción crea una carpeta en una ubicación específica. De forma predeterminada, si es necesario, el módulo también crea de forma recursiva las carpetas principales.

Si la carpeta ya existe en la carpeta especificada, el módulo de acción, de forma predeterminada, trunca o sobrescribe la carpeta existente. Puede anular este comportamiento al configurar la opción sobrescribir como `false`. Cuando la opción de sobreescritura esté establecida como `false` y ya haya una carpeta en la ubicación especificada con el nombre especificado, el módulo de acción devolverá un error.

`owner`, `group`, y `permissions` son entradas opcionales. La entrada para `permissions` debe ser un valor de cadena. Estas opciones no son compatibles con las plataformas Windows. Este módulo de acción valida y devuelve un error si `owner`, `group`, y `permissions` se utilizan en las plataformas Windows.

Este módulo de acción puede crear una carpeta con los permisos definidos por el valor predeterminado `umask` del sistema operativo. Debe establecer el valor `umask` si quiere anular el valor predeterminado.

El módulo de acción devuelve un error cuando ocurre lo siguiente:

- No tiene permiso para crear una carpeta en la ubicación especificada.
- El módulo de acción detecta un error al realizar la operación.

Input

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables	Compatible con todas las plataformas
`path`	La ruta de la carpeta.	Cadena	Sí	N/A	N/A	Sí
`owner`	El nombre o el ID de usuario.	Cadena	No	El usuario actual.	N/A	No es compatible con Windows.
`group`	El nombre o el ID del grupo.	Cadena	No	El grupo del usuario actual.	N/A	No es compatible con Windows.
`permissions`	Permisos para carpetas.	Cadena	No	`0777`	N/A	No es compatible con Windows.
`overwrite`	Si el nombre del archivo especificado ya existe, si se establece este valor como `false` se impide que	Booleano	No	`true`	N/A	Sí

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables	Compatible con todas las plataformas
	el archivo se trunque o sobrescriba de forma predeterminada.					

Ejemplo de entrada: crear una carpeta (Linux)

```
- name: CreatingFolderLinux
  action: CreateFolder
  inputs:
    - path: /Sample/MyFolder/
```

Ejemplo de entrada: crear una carpeta (Windows)

```
- name: CreatingFolderWindows
  action: CreateFolder
  inputs:
    - path: C:\MyFolder
```

Ejemplo de entrada: creación de una carpeta especificando las propiedades de la carpeta

```
- name: CreatingFolderWithFolderProperties
  action: CreateFolder
  inputs:
    - path: /Sample/MyFolder/Sample/
      owner: SampleOwnerName
      group: SampleGroupName
      permissions: 0777
    - path: /Sample/MyFolder/SampleFoler/
      permissions: 777
```

Ejemplo de entrada: cree una carpeta que sobrescriba la carpeta existente, si la hay.

```
- name: CreatingFolderWithOverwrite
  action: CreateFolder
  inputs:
    - path: /Sample/MyFolder/Sample/
      overwrite: true
```

Output

Ninguna.

CreateSymlink (Linux, Windows, macOS)

El módulo de CreateSymlinkacción crea enlaces simbólicos o archivos que contienen una referencia a otro archivo. Este módulo no es compatible con las plataformas Windows.

La entrada de las opciones `path` y `target` puede ser una ruta absoluta o relativa. Si la entrada de la opción `path` es una ruta relativa, se sustituye por la ruta absoluta cuando se crea el enlace.

De forma predeterminada, cuando ya existe un enlace con el nombre especificado en la carpeta especificada, el módulo de acción devuelve un error. Puede anular este comportamiento al configurar la opción `force` como `true`. Si la opción `force` se establece como `true`, el módulo sobrescribirá el enlace existente.

Si no existe una carpeta principal, el módulo de acción crea la carpeta de forma recursiva, de forma predeterminada.

El módulo de acción devuelve un error cuando ocurre lo siguiente:

- El archivo de destino no existe en tiempo de ejecución.
- Ya existe un archivo de enlace no simbólico con el nombre especificado.
- El módulo de acción detecta un error al realizar la operación.

Input

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables	Compatible con todas las plataformas
`path`	La ruta del archivo.	Cadena	Sí	N/A	N/A	No es compatible con Windows.
`target`	La ruta del archivo de destino al que apunta el enlace simbólico.	Cadena	Sí	N/A	N/A	No es compatible con Windows.
`force`	Fuerza la creación de un enlace cuando ya existe un enlace con el mismo nombre.	Booleano	No	`false`	N/A	No es compatible con Windows.

Ejemplo de entrada: creación de un enlace simbólico que obligue a crear un enlace

```
- name: CreatingSymbolicLinkWithForce
  action: CreateSymlink
  inputs:
    - path: /Folder2/Symboliclink.txt
      target: /Folder/Sample.txt
      force: true
```

Ejemplo de entrada: creación de un enlace simbólico que no obligue a crear un enlace

```
- name: CreatingSymbolicLinkWithOutForce
  action: CreateSymlink
  inputs:
    - path: Symboliclink.txt
      target: /Folder/Sample.txt
```

Output

Ninguna.

DeleteFile (Linux, Windows, macOS)

El módulo de DeleteFileacción elimina uno o varios archivos en una ubicación específica.

La entrada de `path` debe ser una ruta de archivo válida o una ruta de archivo con un carácter comodín (*) en el nombre del archivo. Si se especifican caracteres comodín en el nombre del archivo, se eliminarán todos los archivos de la misma carpeta que coincidan con el comodín.

El módulo de acción devuelve un error cuando ocurre lo siguiente:

- No tiene permiso para realizar la operación de eliminación.
- El módulo de acción detecta un error al realizar la operación.

Input

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables	Compatible con todas las plataformas
`path`	La ruta del archivo.	Cadena	Sí	N/A	N/A	Sí

Ejemplo de entrada: eliminación de un solo archivo (Linux)

```
- name: DeletingSingleFileLinux
```

```
    action: DeleteFile
    inputs:
      - path: /SampleFolder/MyFolder/Sample.txt
```

Ejemplo de entrada: eliminación de un solo archivo (Windows)

```
  - name: DeletingSingleFileWindows
    action: DeleteFile
    inputs:
      - path: C:\SampleFolder\MyFolder\Sample.txt
```

Ejemplo de entrada: eliminación de un archivo que termine en “log” (Linux)

```
  - name: DeletingFileEndingWithLogLinux
    action: DeleteFile
    inputs:
      - path: /SampleFolder/MyFolder/*log
```

Ejemplo de entrada: eliminación de un archivo que termine en “log” (Windows)

```
  - name: DeletingFileEndingWithLogWindows
    action: DeleteFile
    inputs:
      - path: C:\SampleFolder\MyFolder\*log
```

Ejemplo de entrada: eliminación de todos los archivos de una carpeta específica (Linux)

```
  - name: DeletingAllFilesInAFolderLinux
    action: DeleteFile
    inputs:
      - path: /SampleFolder/MyFolder/*
```

Ejemplo de entrada: eliminación de todos los archivos de una carpeta específica (Windows)

```
  - name: DeletingAllFilesInAFolderWindows
    action: DeleteFile
    inputs:
      - path: C:\SampleFolder\MyFolder\*
```

Output

Ninguna.

DeleteFolder (Linux, Windows, macOS)

El módulo de DeleteFolderacción elimina las carpetas.

Si la carpeta no está vacía, debe configurar la opción `force` como `true` para eliminar la carpeta y su contenido. Si no establece la `force` opción como `true` y la carpeta que intenta eliminar no está vacía, el módulo de acción mostrará un error. El valor predeterminado de la opción `force` es `false`.

El módulo de acción devuelve un error cuando ocurre lo siguiente:

- No tiene permiso para realizar la operación de eliminación.
- El módulo de acción detecta un error al realizar la operación.

Input

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables	Compatible con todas las plataformas
`path`	La ruta de la carpeta.	Cadena	Sí	N/A	N/A	Sí
`force`	Elimina la carpeta esté vacía o no.	Booleano	No	`false`	N/A	Sí

Ejemplo de entrada: elimine una carpeta que no esté vacía mediante la opción **`force`** (Linux)

```
- name: DeletingFolderWithForceOptionLinux
  action: DeleteFolder
  inputs:
    - path: /Sample/MyFolder/Sample/
      force: true
```

Ejemplo de entrada: elimine una carpeta que no esté vacía mediante la opción **`force`** (Windows)

```
- name: DeletingFolderWithForceOptionWindows
  action: DeleteFolder
  inputs:
    - path: C:\Sample\MyFolder\Sample\
      force: true
```

Ejemplo de entrada: eliminación de una carpeta (Linux)

```
- name: DeletingFolderWithOutForceLinux
  action: DeleteFolder
  inputs:
    - path: /Sample/MyFolder/Sample/
```

Ejemplo de entrada: eliminación de una carpeta (Windows)

```
- name: DeletingFolderWithOutForce
  action: DeleteFolder
  inputs:
    - path: C:\Sample\MyFolder\Sample\
```

Output

Ninguna.

ListFiles (Linux, Windows, macOS)

El módulo de ListFilesacción muestra los archivos de una carpeta específica. Cuando la opción recursiva está establecida como `true`, muestra los archivos en subcarpetas. De forma predeterminada, este módulo no muestra los archivos de las subcarpetas.

Para enumerar todos los archivos con nombres que coincidan con un patrón específico, utilice la opción `fileNamePattern` para proporcionar el patrón. La opción `fileNamePattern` acepta el valor comodín (`*`). Cuando se proporciona el `fileNamePattern`, se devuelven todos los archivos que coincidan con el formato de nombre de archivo especificado.

El módulo de acción devuelve un error cuando ocurre lo siguiente:

- La carpeta especificada no existe en tiempo de ejecución.
- No tiene permiso para crear un archivo o una carpeta en la carpeta especificada.

- El módulo de acción detecta un error al realizar la operación.

Input

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables	Compatible con todas las plataformas
`path`	La ruta de la carpeta.	Cadena	Sí	N/A	N/A	Sí
`fileNamePattern`	El patrón que debe coincidir para mostrar todos los archivos con nombres que coincidan con el patrón.	Cadena	No	N/A	N/A	Sí
`recursive`	Muestra los archivos de la carpeta de forma recursiva.	Booleano	No	`false`	N/A	Sí

Ejemplo de entrada: lista los archivos de la carpeta especificada (Linux)

```
  - name: ListingFilesInSampleFolderLinux
```

```
  action: ListFiles
  inputs:
    - path: /Sample/MyFolder/Sample
```

Ejemplo de entrada: lista los archivos de la carpeta especificada (Windows)

```
- name: ListingFilesInSampleFolderWindows
  action: ListFiles
  inputs:
    - path: C:\Sample\MyFolder\Sample
```

Ejemplo de entrada: lista los archivos que terminan en “log” (Linux)

```
- name: ListingFilesWithEndingWithLogLinux
  action: ListFiles
  inputs:
    - path: /Sample/MyFolder/
      fileNamePattern: *log
```

Ejemplo de entrada: lista los archivos que terminan en “log” (Windows)

```
- name: ListingFilesWithEndingWithLogWindows
  action: ListFiles
  inputs:
    - path: C:\Sample\MyFolder\
      fileNamePattern: *log
```

Ejemplo de entrada: lista los archivos de forma recursiva

```
- name: ListingFilesRecursively
  action: ListFiles
  inputs:
    - path: /Sample/MyFolder/
      recursive: true
```

Output

Nombre de la clave	Descripción	Tipo
`files`	La lista de archivos.	Cadena

Ejemplo de resultados

```
{
 "files": "/sample1.txt,/sample2.txt,/sample3.txt"
}
```

MoveFile (Linux, Windows, macOS)

El módulo de MoveFileacción mueve los archivos del origen especificado al destino especificado.

Si el archivo ya existe en la carpeta especificada, el módulo de acción sobrescribe el archivo existente de forma predeterminada. Puede anular este comportamiento al configurar la opción sobrescribir como `false`. Cuando la opción de sobreescritura esté establecida como `false` y ya haya un archivo en la ubicación especificada con el nombre especificado, el módulo de acción devolverá un error. Esta opción funciona igual que el comando de Linux `mv`, que se sobrescribe de forma predeterminada.

El nombre del archivo fuente puede incluir un comodín (*). Los caracteres comodín solo se aceptan después del último separador de ruta del archivo (/ o \). Si se incluyen caracteres comodín en el nombre del archivo de origen, todos los archivos que coincidan con el comodín se copian en la carpeta de destino. Si desea mover más de un archivo con un carácter comodín, la entrada de la opción `destination` debe terminar con un separador de rutas de archivo (/ o \), lo que indica que la entrada de destino es una carpeta.

Si el nombre del archivo de destino es diferente del nombre del archivo de origen, puede especificar el nombre del archivo de destino mediante la opción `destination`. Si no especifica un nombre de archivo de destino, se utilizará el nombre del archivo de origen para crear el archivo de destino. Cualquier texto que siga al separador (/ o \) de la última ruta del archivo se trata como el nombre del archivo. Si desea utilizar el mismo nombre de archivo que el archivo de origen, la entrada de la opción `destination` debe terminar con un separador de rutas de archivo (/ o \).

El módulo de acción devuelve un error cuando ocurre lo siguiente:

- No tiene permiso para crear un archivo en la carpeta especificada.
- Los archivos de origen no existen en tiempo de ejecución.
- Ya existe una carpeta con el nombre de archivo especificado y la opción `overwrite` está configurada como `false`.
- El módulo de acción detecta un error al realizar la operación.

Input

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables	Compatible con todas las plataformas
`source`	La ruta del archivo de origen.	Cadena	Sí	N/A	N/A	Sí
`destination`	La ruta del archivo de destino.	Cadena	Sí	N/A	N/A	Sí
`overwrite`	Si se establece como falso, los archivos de destino no se reemplazarán cuando ya haya un archivo en la ubicación	Booleano	No	`true`	N/A	Sí

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables	Compatible con todas las plataformas
	especificada con el nombre especificado.					

Ejemplo de entrada: mover un archivo (Linux)

```
- name: MovingAFileLinux
  action: MoveFile
  inputs:
    - source: /Sample/MyFolder/Sample.txt
      destination: /MyFolder/destinationFile.txt
```

Ejemplo de entrada: mover un archivo (Windows)

```
- name: MovingAFileWindows
  action: MoveFile
  inputs:
    - source: C:\Sample\MyFolder\Sample.txt
      destination: C:\MyFolder\destinationFile.txt
```

Ejemplo de entrada: mueva un archivo utilizando el nombre del archivo fuente (Linux)

```
- name: MovingFileWithSourceFileNameLinux
  action: MoveFile
  inputs:
    - source: /Sample/MyFolder/Sample.txt
      destination: /MyFolder/
```

Ejemplo de entrada: mueva un archivo con el nombre del archivo fuente (Windows)

```
- name: MovingFileWithSourceFileNameWindows
```

```
  action: MoveFile
  inputs:
    - source: C:\Sample\MyFolder\Sample.txt
      destination: C:\MyFolder
```

Ejemplo de entrada: mover un archivo con un carácter comodín (Linux)

```
- name: MovingFilesWithWildCardLinux
  action: MoveFile
  inputs:
    - source: /Sample/MyFolder/Sample*
      destination: /MyFolder/
```

Ejemplo de entrada: mover un archivo con un carácter comodín (Windows)

```
- name: MovingFilesWithWildCardWindows
  action: MoveFile
  inputs:
    - source: C:\Sample\MyFolder\Sample*
      destination: C:\MyFolder
```

Ejemplo de entrada: mover un archivo sin sobrescribirlo (Linux)

```
- name: MovingFilesWithoutOverwriteLinux
  action: MoveFile
  inputs:
    - source: /Sample/MyFolder/Sample.txt
      destination: /MyFolder/destinationFile.txt
      overwrite: false
```

Ejemplo de entrada: mover un archivo sin sobrescribirlo (Windows)

```
- name: MovingFilesWithoutOverwrite
  action: MoveFile
  inputs:
    - source: C:\Sample\MyFolder\Sample.txt
      destination: C:\MyFolder\destinationFile.txt
      overwrite: false
```

Output

Ninguna.

MoveFolder (Linux, Windows, macOS)

El módulo de MoveFolderacción mueve las carpetas del origen especificado al destino especificado. La entrada de la opción `source` es la carpeta que se va a mover y la entrada de la opción `destination` es la carpeta en la que se mueve el contenido de la carpeta de origen.

Si la carpeta principal de destino o la entrada de la opción `destination` no existen en tiempo de ejecución, el comportamiento predeterminado del módulo es crear la carpeta de forma recursiva en el destino especificado.

Si ya existe una carpeta con el nombre especificado en la carpeta especificada, el módulo de acción, de forma predeterminada, sobrescribe la carpeta existente. Puede anular este comportamiento al configurar la opción sobrescribir como `false`. Cuando la opción de sobreescritura esté establecida como `false` y ya haya una carpeta en la ubicación especificada con el nombre especificado, el módulo de acción devolverá un error.

El nombre de la carpeta fuente puede incluir un comodín (`*`). Los caracteres comodín solo se aceptan después del último separador de ruta del archivo (`/` o `\`). Si se incluyen caracteres comodín en el nombre de la carpeta de origen, todas las carpetas que coincidan con el comodín se copian en la carpeta de destino. Si desea mover más de una carpeta con un carácter comodín, la entrada de la opción `destination` debe terminar con un separador de rutas de archivos (`/` o `\`), lo que indica que la entrada de destino es una carpeta.

Si el nombre de la carpeta de destino es diferente del nombre de la carpeta de origen, puede especificar el nombre de la carpeta de destino mediante la opción `destination`. Si no especifica un nombre de carpeta de destino, se utilizará el nombre de la carpeta de origen para crear la carpeta de destino. Cualquier texto que siga al separador (`/` o `\`) de la última ruta del archivo se trata como el nombre de la carpeta. Si desea utilizar el mismo nombre de carpeta que la carpeta de origen, la entrada de la opción `destination` debe terminar con un separador de rutas de archivo (`/` o `\`).

El módulo de acción devuelve un error cuando ocurre lo siguiente:

- No tiene permiso para crear una carpeta en la carpeta de destino.
- Las carpetas de origen no existen en tiempo de ejecución.
- Ya existe una carpeta con el nombre especificado y la opción `overwrite` está configurada como `false`.
- El módulo de acción detecta un error al realizar la operación.

Input

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables	Compatible con todas las plataformas
`source`	La ruta de la carpeta de origen.	Cadena	Sí	N/A	N/A	Sí
`destination`	La ruta de la carpeta de destino.	Cadena	Sí	N/A	N/A	Sí
`overwrite`	Si se establece como falso, las carpetas de destino no se reemplazarán cuando ya haya una carpeta en la ubicación especificada con el nombre especificado.	Booleano	No	`true`	N/A	Sí

Ejemplo de entrada: mover una carpeta (Linux)

```
- name: MovingAFolderLinux
  action: MoveFolder
  inputs:
    - source: /Sample/MyFolder/SourceFolder
      destination: /MyFolder/destinationFolder
```

Ejemplo de entrada: mover una carpeta (Windows)

```
- name: MovingAFolderWindows
  action: MoveFolder
  inputs:
    - source: C:\Sample\MyFolder\SourceFolder
      destination: C:\MyFolder\destinationFolder
```

Ejemplo de entrada: mover una carpeta con el nombre de la carpeta de origen (Linux)

```
- name: MovingFolderWithSourceFolderNameLinux
  action: MoveFolder
  inputs:
    - source: /Sample/MyFolder/SampleFolder
      destination: /MyFolder/
```

Ejemplo de entrada: mover una carpeta con el nombre de la carpeta de origen (Windows)

```
- name: MovingFolderWithSourceFolderNameWindows
  action: MoveFolder
  inputs:
    - source: C:\Sample\MyFolder\SampleFolder
      destination: C:\MyFolder\
```

Ejemplo de entrada: mover una carpeta con un carácter comodín (Linux)

```
- name: MovingFoldersWithWildCardLinux
  action: MoveFolder
  inputs:
    - source: /Sample/MyFolder/Sample*
      destination: /MyFolder/
```

Ejemplo de entrada: mover una carpeta con un carácter comodín (Windows)

```
- name: MovingFoldersWithWildCardWindows
```

```
  action: MoveFolder
  inputs:
    - source: C:\Sample\MyFolder\Sample*
      destination: C:\MyFolder\
```

Ejemplo de entrada: mover una carpeta sin sobrescribirla (Linux)

```
- name: MovingFoldersWithoutOverwriteLinux
  action: MoveFolder
  inputs:
    - source: /Sample/MyFolder/SampleFolder
  destination: /MyFolder/destinationFolder
  overwrite: false
```

Ejemplo de entrada: mover una carpeta sin sobrescribirla (Windows)

```
- name: MovingFoldersWithoutOverwriteWindows
  action: MoveFolder
  inputs:
    - source: C:\Sample\MyFolder\SampleFolder
      destination: C:\MyFolder\destinationFolder
      overwrite: false
```

Output

Ninguna.

ReadFile (Linux, Windows, macOS)

El módulo de ReadFileacción lee el contenido de un archivo de texto de tipo cadena. Este módulo se puede utilizar para leer el contenido de un archivo y utilizarlo en los pasos siguientes mediante el encadenamiento o para leer los datos del archivo `console.log`. Si la ruta especificada es un enlace simbólico, este módulo devuelve el contenido del archivo de destino. Este módulo solo admite archivos de texto.

Si el valor de codificación del archivo es diferente del valor de codificación (`utf-8`) predeterminado, puede especificar el valor de codificación del archivo mediante la opción `encoding`. De forma predeterminada, se supone que `utf-16` y `utf-32` utilizan la codificación little-endian.

De forma predeterminada, este módulo no puede imprimir el contenido del archivo en el archivo. `console.log` Puede anular esta configuración configurando la propiedad `printFileContent` como `true`.

Este módulo solo puede devolver el contenido de un archivo. No puede analizar archivos, como los archivos Excel o JSON.

El módulo de acción devuelve un error cuando ocurre lo siguiente:

- El archivo no existe en el tiempo de ejecución.
- El módulo de acción detecta un error al realizar la operación.

Input

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables	Compatible con todas las plataformas
`path`	La ruta del archivo.	Cadena	Sí	N/A	N/A	Sí
`encoding`	El estándar de la codificación.	Cadena	No	`utf8`	`utf8`, `utf-8`, `utf16`,`utf-16`, `utf16-LE`, `utf-16-LE` `utf16-BE`, `utf-16-BE` , `utf32`, `utf-32`, `utf32-LE`,`utf-32-LE` , `utf32-BE`, y `utf-32-BE` . El valor de la	Sí

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables	Compatible con todas las plataformas
					opción de codificación no distingue entre mayúsculas y minúsculas.	
`printFileContent`	Imprime el contenido del archivo en el archivo `console.log` .	Booleano	No	false	N/A	Sí.

Ejemplo de entrada: leer un archivo (Linux)

```
- name: ReadingFileLinux
  action: ReadFile
  inputs:
    - path: /home/UserName/SampleFile.txt
```

Ejemplo de entrada: leer un archivo (Windows)

```
- name: ReadingFileWindows
  action: ReadFile
  inputs:
    - path: C:\Windows\WindowsUpdate.log
```

Ejemplo de entrada: cómo leer un archivo y especificar el estándar de codificación

```
- name: ReadingFileWithFileEncoding
  action: ReadFile
  inputs:
    - path: /FolderName/SampleFile.txt
      encoding: UTF-32
```

Ejemplo de entrada: cómo leer un archivo e imprimirlo en el archivo **`console.log`**

```
- name: ReadingFileToConsole
  action: ReadFile
  inputs:
    - path: /home/UserName/SampleFile.txt
      printFileContent: true
```

Output

Campo	Descripción	Tipo
`content`	El contenido del archivo.	cadena

Ejemplo de resultados

```
{
 "content" : "The file content"
}
```

SetFileEncoding (Linux, Windows, macOS)

El módulo de SetFileEncodingacción modifica la propiedad de codificación de un archivo existente. Este módulo puede convertir la codificación de archivos `utf-8` a un estándar de codificación específico. De forma predeterminada, `utf-16` y `utf-32` se supone que utilizan la codificación little-endian.

El módulo de acción devuelve un error cuando ocurre lo siguiente:

- No tiene permiso para realizar la modificación especificada.
- El archivo no existe en el tiempo de ejecución.

- El módulo de acción detecta un error al realizar la operación.

Input

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables	Compatible con todas las plataformas
`path`	La ruta del archivo.	Cadena	Sí	N/A	N/A	Sí
`encoding`	El estándar de la codificación.	Cadena	No	`utf8`	`utf8`, `utf-8`, `utf16`,`utf-16`, `utf16-LE`, `utf-16-LE` `utf16-BE`, `utf-16-BE` , `utf32`, `utf-32`, `utf32-LE`,`utf-32-LE` , `utf32-BE`, y `utf-32-BE` . El valor de la opción de codificación no distingue entre	Sí

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables	Compatible con todas las plataformas
					mayúsculas y minúsculas.	

Ejemplo de entrada: cómo establecer la propiedad de codificación del archivo

```
- name: SettingFileEncodingProperty
  action: SetFileEncoding
  inputs:
    - path: /home/UserName/SampleFile.txt
      encoding: UTF-16
```

Output

Ninguna.

SetFileOwner (Linux, Windows, macOS)

El módulo de SetFileOwneracción modifica las propiedades de `group` propietario `owner` y las propiedades de un archivo existente. Si el archivo especificado es un enlace simbólico, el módulo modifica la propiedad `owner` del archivo fuente. Este módulo no es compatible con las plataformas Windows.

Este módulo acepta nombres de usuarios y grupos como entradas. Si no se proporciona el nombre del grupo, el módulo asigna el propietario del archivo al grupo al que pertenece el usuario.

El módulo de acción devuelve un error cuando ocurre lo siguiente:

- No tiene permiso para realizar la modificación especificada.
- El nombre de usuario o grupo especificado no existe en el tiempo de ejecución.
- El archivo no existe en el tiempo de ejecución.
- El módulo de acción detecta un error al realizar la operación.

Input

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables	Compatible con todas las plataformas
`path`	La ruta del archivo.	Cadena	Sí	N/A	N/A	No es compatible con Windows.
`owner`	El nombre de usuario.	cadena	Sí	N/A	N/A	No es compatible con Windows.
`group`	El nombre del grupo de usuarios.	Cadena	No	El nombre del grupo al que pertenece el usuario.	N/A	No es compatible con Windows.

Ejemplo de entrada: cómo establecer la propiedad del propietario del archivo sin especificar el nombre del grupo de usuarios

```
- name: SettingFileOwnerPropertyNoGroup
  action: SetFileOwner
  inputs:
    - path: /home/UserName/SampleText.txt
      owner: LinuxUser
```

Ejemplo de entrada: establezca la propiedad del propietario del archivo especificando el propietario y el grupo de usuarios

```
- name: SettingFileOwnerProperty
  action: SetFileOwner
  inputs:
```

```
      - path: /home/UserName/SampleText.txt
        owner: LinuxUser
        group: LinuxUserGroup
```

Output

Ninguna.

SetFolderOwner (Linux, Windows, macOS)

El módulo de SetFolderOwneracción modifica de forma recursiva las propiedades del `group` propietario `owner` y las de los propietarios de una carpeta existente. De forma predeterminada, el módulo puede modificar la propiedad de todo el contenido de una carpeta. Para omitir este comportamiento, se puede configurar la opción `recursive` como `false`. Este módulo no es compatible con las plataformas Windows.

Este módulo acepta nombres de usuarios y grupos como entradas. Si no se proporciona el nombre del grupo, el módulo asigna el propietario del archivo al grupo al que pertenece el usuario.

El módulo de acción devuelve un error cuando ocurre lo siguiente:

- No tiene permiso para realizar la modificación especificada.
- El nombre de usuario o grupo especificado no existe en el tiempo de ejecución.
- La carpeta no existe en el tiempo de ejecución.
- El módulo de acción detecta un error al realizar la operación.

Input

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables	Compatible con todas las plataformas
`path`	La ruta de la carpeta.	Cadena	Sí	N/A	N/A	No es compatible con Windows.

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables	Compatible con todas las plataformas
`owner`	El nombre de usuario.	cadena	Sí	N/A	N/A	No es compatible con Windows.
`group`	El nombre del grupo de usuarios.	Cadena	No	El nombre del grupo al que pertenece el usuario.	N/A	No es compatible con Windows.
`recursive`	Anula el comportamiento predeterminado de modificar la propiedad de todo el contenido de una carpeta cuando se establece como `false`.	Booleano	No	`true`	N/A	No es compatible con Windows.

Ejemplo de entrada: cómo establecer la propiedad del propietario de la carpeta sin especificar el nombre del grupo de usuarios

```
  - name: SettingFolderPropertyWithOutGroup
```

```
    action: SetFolderOwner
    inputs:
      - path: /SampleFolder/
        owner: LinuxUser
```

Ejemplo de entrada: cómo establecer la propiedad del propietario de la carpeta sin anular la propiedad de todo el contenido de una carpeta

```
- name: SettingFolderPropertyWithOutRecursively
  action: SetFolderOwner
  inputs:
    - path: /SampleFolder/
      owner: LinuxUser
      recursive: false
```

Ejemplo de entrada: cómo establecer la propiedad del propietario del archivo especificando el nombre del grupo de usuarios

```
- name: SettingFolderPropertyWithGroup
  action: SetFolderOwner
  inputs:
    - path: /SampleFolder/
      owner: LinuxUser
      group: LinuxUserGroup
```

Output

Ninguna.

SetFilePermissions (Linux, Windows, macOS)

El módulo de SetFilePermissionsacción modifica el `permissions` de un archivo existente. Este módulo no es compatible con las plataformas Windows.

La entrada para `permissions` debe ser un valor de cadena.

Este módulo de acción puede crear un archivo con los permisos definidos por el valor umask predeterminado del sistema operativo. Debe establecer el valor `umask` si quiere anular el valor predeterminado.

El módulo de acción devuelve un error cuando ocurre lo siguiente:

- No tiene permiso para realizar la modificación especificada.
- El archivo no existe en el tiempo de ejecución.
- El módulo de acción detecta un error al realizar la operación.

Input

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables	Compatible con todas las plataformas
`path`	La ruta del archivo.	Cadena	Sí	N/A	N/A	No es compatible con Windows.
`permissions`	Permisos de archivos.	Cadena	Sí	N/A	N/A	No es compatible con Windows.

Ejemplo de entrada: modificación de los permisos de los archivos

```
- name: ModifyingFilePermissions
  action: SetFilePermissions
  inputs:
    - path: /home/UserName/SampleFile.txt
      permissions: 766
```

Output

Ninguna.

SetFolderPermissions (Linux, Windows, macOS)

El módulo de SetFolderPermissionsacción modifica de forma recursiva `permissions` la carpeta existente y todos sus subarchivos y subcarpetas. De forma predeterminada, este módulo puede

modificar los permisos para todo el contenido de la carpeta especificada. Para omitir este comportamiento, se puede configurar la opción `recursive` como `false`. Este módulo no es compatible con las plataformas Windows.

La entrada para `permissions` debe ser un valor de cadena.

Este módulo de acción puede modificar los permisos según el valor de umask predeterminado del sistema operativo. Debe establecer el valor `umask` si quiere anular el valor predeterminado.

El módulo de acción devuelve un error cuando ocurre lo siguiente:

- No tiene permiso para realizar la modificación especificada.
- La carpeta no existe en el tiempo de ejecución.
- El módulo de acción detecta un error al realizar la operación.

Input

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables	Compatible con todas las plataformas
`path`	La ruta de la carpeta.	Cadena	Sí	N/A	N/A	No es compatible con Windows.
`permissions`	Permisos para carpetas.	Cadena	Sí	N/A	N/A	No es compatible con Windows.
`recursive`	Anula el comportamiento predeterminado de modificar	Booleano	No	`true`	N/A	No es compatible con Windows.

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables	Compatible con todas las plataformas
	los permisos de todo el contenido de una carpeta cuando se establece como `false`.					

Ejemplo de entrada: establecimiento de permisos de carpeta

```
- name: SettingFolderPermissions
  action: SetFolderPermissions
  inputs:
    - path: SampleFolder/
      permissions: 0777
```

Ejemplo de entrada: cómo establecer los permisos de una carpeta sin modificar los permisos para todo el contenido de una carpeta

```
- name: SettingFolderPermissionsNoRecursive
  action: SetFolderPermissions
  inputs:
    - path: /home/UserName/SampleFolder/
      permissions: 777
      recursive: false
```

Output

Ninguna.

Acciones de instalación de software

En la siguiente sección se describen los módulos de acción que instalan o desinstalan software.

Requisitos de IAM

Si la ruta de descarga de la instalación es un URI de S3, el rol de IAM que asocie al perfil de instancia debe tener permiso para ejecutar el módulo de acción `S3Download`. Para conceder el permiso necesario, adjunte la política de IAM `S3:GetObject` al rol de IAM asociado a su perfil de instancia y especifique la ruta de acceso a su bucket. Por ejemplo, *`arn:aws:s3:::BucketName/*`*).

Entradas MSI complejas

Si las cadenas de entrada contienen caracteres entre comillas dobles (''), debe utilizar uno de los siguientes métodos para asegurarse de que se interpretan correctamente:

- Puede utilizar comillas simples (') en la parte exterior de la cadena para contenerla y comillas dobles (") dentro de la cadena, como se muestra en el siguiente ejemplo.

  ```
  properties:
    COMPANYNAME: '"Acme ""Widgets"" and ""Gizmos."""'
  ```

  En este caso, si necesita usar un apóstrofo dentro de la cadena, debe evitarlo. Esto significa usar otra comilla simple (') antes del apóstrofo.
- Puede usar comillas dobles (") en la parte exterior de la cadena para contenerla. Además, puede evitar las comillas dobles que haya dentro de la cadena utilizando el carácter de barra invertida (\), como se muestra en el siguiente ejemplo.

  ```
  properties:
    COMPANYNAME: "\"Acme \"\"Widgets\"\" and \"\"Gizmos.\"\"\""
  ```

Ambos métodos pasan el valor `COMPANYNAME="Acme ""Widgets"" and ""Gizmos."""` al comando msiexec.

Módulos de acción de instalación de software

- InstallMSI (Windows)
- UninstallMSI (Windows)

InstallMSI (Windows)

El módulo de acción `InstallMSI` instala una aplicación de Windows mediante un archivo MSI. Puede especificar el archivo MSI mediante una ruta local, un URI de objeto S3 o una URL web. La opción de reinicio configura el comportamiento de reinicio del sistema.

TOE de AWS genera el msiexec comando en función de los parámetros de entrada del módulo de acción. Los valores de los parámetros de entrada `path` (ubicación del archivo MSI) y `logFile` (ubicación del archivo de registro) deben escribirse entre comillas (").

Los siguientes códigos de salida MSI se consideran correctos:

- 0 (Success)
- 1614 (ERROR_PRODUCT_UNINSTALLED)
- 1641 (reinicio iniciado)
- 3010 (es necesario reiniciar)

Input

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables
`path`	Especifique la ubicación del archivo MSI, usando una de las siguientes opciones: • La ruta de archivo local. La ruta puede ser	Cadena	Sí	N/A	N/A

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables
	absoluta o relativa. • Un URI de objeto S3 válido. • Una HTTP/HTTPS URL web válida (se recomienda HTTPS) que siga el estándar RFC 3986. Se permite el encadenamiento de expresiones.				

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables
`reboot`	Configure el comportamiento de reinicio del sistema después de que el módulo de acción se ejecute correctamente. Configuración: • `Force`: inicia un reinicio del sistema después de que el comando msiexec se ejecute correctamente. • `Allow`: inicia un reinicio del sistema si	Cadena	No	`Allow`	`Allow`, `Force`, `Skip`

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables
	el comando msiexec devuelve un código de salida que indica que es necesario reiniciar el sistema. • `Skip`: registra un mensaje informativo en el archivo `console.log` que indica que se ha omitido un reinicio. Esta opción impide el reinicio, incluso si el comando msiexec devuelve un código de salida				

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables
	que indica que es necesario reiniciar.				

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables
`logOptions`	Especifique las opciones que se van a utilizar para el registro de la instalación de MSI. Los indicadores especificados se pasan al instalador de MSI, junto con el parámetro de línea de comandos `/L` para habilitar el registro. Si no se especifica ningún indicador, TOE de AWS utiliza el valor predeterminado. Para obtener más información acerca de opciones	Cadena	No	`*VX`	`i,w,e,a,r,u,c,m,o,p,v,x,+,!,*`

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables
	de registro MSI, consulte Opciones de línea de comandos en la documentación de del producto Windows Installer Microsoft.				
`logFile`	Una ruta absoluta o relativa a la ubicación del archivo de registro. Si la ruta del archivo de registro no existe, créela. Si no se proporciona la ruta del archivo de registro, TOE de AWS no almacena el registro de instalación de MSI.	Cadena	No	N/A	N/A

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables
`properties`	Los pares clave-valor de las propiedades de registro MSI, por ejemplo, `TARGETDIR: "C:\target\location"` Nota: No se permite la modificación de las siguientes propiedades: • `REBOOT="ReallySuppress"` • `REINSTALLMODE="ecmus"` • `REINSTALL="ALL"`	Map[String]String	No	N/A	N/A

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables
`ignoreAuthenticodeSignatureErrors`	Marcador para ignorar los errores de validación de la firma de Authenticode para el instalador especificado en la ruta. El comando Get-AuthenticodeSignature se utiliza para validar los instaladores. Configuración: • `true`: los errores de validación se ignoran y el instalador se ejecuta. • `false`: los errores de validació	Booleano	No	`false`	`true`, `false`

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables
	n no se ignoran. El instalador solo se ejecuta cuando la validación se realiza correctamente. Este es el comportamiento predeterminado.				

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables
`allowUnsignedInstaller`	Marca que permite ejecutar el instalador sin firma especificado en la ruta. El comando Get-AuthenticodeSignature se utiliza para validar los instaladores. Configuración: • `true`: ignora el estado `NotSigned` devuelto por el comando Get-AuthenticodeSignature y ejecuta el instalador. •	Booleano	No	`false`	`true`, `false`

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables
	`false`: requiere que el instalador esté firmado. Los instaladores sin firma no se ejecutarán. Este es el comportamiento predeterminado.				

Ejemplos

Los ejemplos siguientes muestran variaciones de la sección de entrada de su documento de componentes, en función de la ruta de instalación.

Ejemplo de entrada: instalación de la ruta del documento local

```
- name: local-path-install
  steps:
    - name: LocalPathInstaller
      action: InstallMSI
      inputs:
        path: C:\sample.msi
        logFile: C:\msilogs\local-path-install.log
        logOptions: '*VX'
        reboot: Allow
        properties:
```

```
          COMPANYNAME: '"Amazon Web Services"'
        ignoreAuthenticodeSignatureErrors: true
        allowUnsignedInstaller: true
```

Ejemplo de entrada: instalación de la ruta Amazon S3

```
- name: s3-path-install
  steps:
    - name: S3PathInstaller
      action: InstallMSI
      inputs:
        path: s3://<bucket-name>/sample.msi
        logFile: s3-path-install.log
        reboot: Force
        ignoreAuthenticodeSignatureErrors: false
        allowUnsignedInstaller: true
```

Ejemplo de entrada: instalación de una ruta web

```
- name: web-path-install
  steps:
    - name: WebPathInstaller
      action: InstallMSI
      inputs:
        path: https://<some-path>/sample.msi
        logFile: web-path-install.log
        reboot: Skip
        ignoreAuthenticodeSignatureErrors: true
        allowUnsignedInstaller: false
```

Output

Véase a continuación un ejemplo de la salida del módulo de acción `InstallMSI`.

```
{
 "logFile": "web-path-install.log",
 "msiExitCode": 0,
 "stdout": ""
}
```

UninstallMSI (Windows)

Este módulo de acción `UninstallMSI` le permite desinstalar una aplicación de Windows utilizando un archivo MSI. Puede especificar la ubicación del archivo MSI mediante una ruta de archivo local, un URI de objeto S3 o una URL web. La opción de reinicio configura el comportamiento de reinicio del sistema.

TOE de AWS genera el msiexec comando en función de los parámetros de entrada del módulo de acción. La ubicación del archivo MSI (`path`) y la ubicación del archivo de registro (`logFile`) se escriben explícitamente entre comillas dobles (") al generar el comando msiexec.

Los siguientes códigos de salida MSI se consideran correctos:

- 0 (Success)
- 1605 (ERROR_UNKNOWN_PRODUCT)
- 1614 (ERROR_PRODUCT_UNINSTALLED)
- 1641 (reinicio iniciado)
- 3010 (es necesario reiniciar)

Input

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables
`path`	Especifique la ubicación del archivo MSI, usando una de las siguientes opciones: • La ruta de archivo local. La ruta	Cadena	Sí	N/A	N/A

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables
	puede ser absoluta o relativa. • Un URI de objeto S3 válido. • Una HTTP/HTTPS URL web válida (se recomienda HTTPS) que siga el estándar RFC 3986. Se permite el encadenamiento de expresiones.				

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables
`reboot`	Configura el comportamiento de reinicio del sistema después de que el módulo de acción se ejecute correctamente. Configuración: • `Force`: inicia un reinicio del sistema después de que el comando msiexec se ejecute correctamente. • `Allow`: inicia un reinicio del sistema si	Cadena	No	`Allow`	`Allow`, `Force`, `Skip`

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables
	el comando msiexec devuelve un código de salida que indica que es necesario reiniciar el sistema. • `Skip`: registra un mensaje informativo en el archivo `console.log` que indica que se ha omitido un reinicio. Esta opción impide el reinicio, incluso si el comando msiexec devuelve un código de salida				

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables
	que indica que es necesario reiniciar.				

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables
`logOptions`	Especifique las opciones que se van a utilizar para el registro de la instalación de MSI. Los indicadores especificados se pasan al instalador de MSI, junto con el parámetro de línea de comandos `/L` para habilitar el registro. Si no se especifica ningún indicador, TOE de AWS utiliza el valor predeterminado. Para obtener más información acerca de opciones	Cadena	No	`*VX`	`i,w,e,a,r,u,c,m,o,p,v,x,+,!,*`

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables
	de registro MSI, consulte Opciones de línea de comandos en la documentación de del producto Windows Installer Microsoft.				
`logFile`	Una ruta absoluta o relativa a la ubicación del archivo de registro. Si la ruta del archivo de registro no existe, créela. Si no se proporciona la ruta del archivo de registro, TOE de AWS no almacena el registro de instalación de MSI.	Cadena	No	N/A	N/A

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables
`properties`	Los pares clave-valor de las propiedades de registro MSI, por ejemplo, `TARGETDIR: "C:\target\location"` Nota: No se permite la modificación de las siguientes propiedades: • `REBOOT="ReallySuppress"` • `REINSTALLMODE="ecmus"` • `REINSTALL="ALL"`	Map[String]String	No	N/A	N/A

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables
`ignoreAuthenticodeSignatureErrors`	Marcador para ignorar los errores de validación de la firma de Authenticode para el instalador especificado en la ruta. El comando Get-AuthenticodeSignature se utiliza para validar los instaladores. Configuración: • `true`: los errores de validación se ignoran y el instalador se ejecuta. • `false`: los errores de validació	Booleano	No	`false`	`true`, `false`

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables
	n no se ignoran. El instalador solo se ejecuta cuando la validación se realiza correctamente. Este es el comportamiento predeterminado.				

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables
`allowUnsignedInstaller`	Marca que permite ejecutar el instalador sin firma especificado en la ruta. El comando Get-AuthenticodeSignature se utiliza para validar los instaladores. Configuración: • `true`: ignora el estado `NotSigned` devuelto por el comando Get-AuthenticodeSignature y ejecuta el instalador. •	Booleano	No	`false`	`true`, `false`

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Valor predeterminado	Valores aceptables
	`false`: requiere que el instalador esté firmado. Los instaladores sin firma no se ejecutarán. Este es el comportamiento predeterminado.				

Ejemplos

Los ejemplos siguientes muestran variaciones de la sección de entrada de su documento de componentes, en función de la ruta de instalación.

Ejemplo de entrada: eliminación de la instalación de la ruta del documento local

```
- name: local-path-uninstall
  steps:
    - name: LocalPathUninstaller
      action: UninstallMSI
      inputs:
        path: C:\sample.msi
        logFile: C:\msilogs\local-path-uninstall.log
        logOptions: '*VX'
        reboot: Allow
        properties:
```

```
          COMPANYNAME: '"Amazon Web Services"'
        ignoreAuthenticodeSignatureErrors: true
        allowUnsignedInstaller: true
```

Ejemplo de entrada: eliminación de la instalación de la ruta Amazon S3

```
- name: s3-path-uninstall
  steps:
    - name: S3PathUninstaller
      action: UninstallMSI
      inputs:
        path: s3://<bucket-name>/sample.msi
        logFile: s3-path-uninstall.log
        reboot: Force
        ignoreAuthenticodeSignatureErrors: false
        allowUnsignedInstaller: true
```

Ejemplo de entrada: eliminación de la instalación de una ruta web

```
- name: web-path-uninstall
  steps:
    - name: WebPathUninstaller
      action: UninstallMSI
      inputs:
        path: https://<some-path>/sample.msi
        logFile: web-path-uninstall.log
        reboot: Skip
        ignoreAuthenticodeSignatureErrors: true
        allowUnsignedInstaller: false
```

Output

Véase a continuación un ejemplo de la salida del módulo de acción `UninstallMSI`.

```
{
 "logFile": "web-path-uninstall.log",
 "msiExitCode": 0,
 "stdout": ""
}
```

Módulos de acción del sistema

En la siguiente sección se describen los módulos de acción que ejecutan acciones del sistema o actualizan la configuración del sistema.

Módulos de acción del sistema

- Reboot (Linux, Windows)
- SetRegistry (Windows)
- UpdateOS (Linux, Windows)

Reboot (Linux, Windows)

El módulo de acción Reboot reinicia la instancia. Tiene una opción configurable para retrasar el inicio del reinicio. De forma predeterminada, `delaySeconds` está configurada como `0`, lo que significa que no hay ningún retraso. El tiempo de espera por pasos no es compatible con el módulo de acción de reinicio, ya que no se aplica cuando se reinicia la instancia.

Si Systems Manager Agent invoca la aplicación, entrega el código de salida (`3010` para Windows o `194` Linux) a Systems Manager Agent. Systems Manager Agent gestiona el reinicio del sistema tal y como se describe en Reiniciar una instancia gestionada desde scripts.

Si la aplicación se invoca en el host como un proceso independiente, guarda el estado de ejecución actual, configura un activador de ejecución automática posterior al reinicio para volver a ejecutar la aplicación después del reinicio y, a continuación, reinicia el sistema.

Activador de ejecución automática posterior al reinicio:

- Windows. TOE de AWS crea una entrada del programador de tareas de Windows con un activador que se ejecuta automáticamente en `SystemStartup`
- Linux. TOE de AWS agrega un trabajo en crontab que se ejecuta automáticamente después de que el sistema se reinicie.

```
@reboot /download/path/awstoe run --document s3://bucket/key/doc.yaml
```

Este activador se borra cuando se inicia la aplicación.

Reintentos

De forma predeterminada, el número máximo de reintentos se establece en `CommandRetryLimit` de Systems Manager. Si el número de reinicios supera el límite de reintentos, se produce un error en la automatización. Para cambiar el límite, puede editar el archivo de configuración del agente de Systems Manager (`Mds.CommandRetryLimit`). Consulte Runtime Configuration en el código abierto del agente de Systems Manager.

Para utilizar el módulo de acción reinicio, para los pasos que contienen el reinicio `exitcode` (por ejemplo, `3010`), debe ejecutar la aplicación binaria como `sudo user`.

Input

Nombre de la clave	Descripción	Tipo	Obligatorio/a	Predeterminado/a
`delaySeconds`	Retrasa un tiempo específico antes de iniciar un reinicio.	Entero	No	`0`

Ejemplo de entrada: paso de reinicio

```
- name: RebootStep
  action: Reboot
  onFailure: Abort
  maxAttempts: 2
  inputs:
    delaySeconds: 60
```

Salida

Ninguna.

Cuando finalice el módulo Reboot, Generador de Imágenes continúa con el siguiente paso de la compilación.

SetRegistry (Windows)

El módulo de SetRegistryacción acepta una lista de entradas y permite establecer el valor de la clave de registro especificada. Si no existe una clave de registro, se crea en la ruta definida. Esta característica solo se aplica a los clientes de Windows.

Input

Nombre de la clave	Descripción	Tipo	Obligatorio/a
`path`	Ruta de la clave de registro.	Cadena	Sí
`name`	Nombre de la clave de registro.	Cadena	Sí
`value`	Valor de la clave de registro.	String/Number/Array	Sí
`type`	Valor de la clave de registro.	Cadena	Sí

Prefijos de ruta admitidos

- `HKEY_CLASSES_ROOT / HKCR:`
- `HKEY_USERS / HKU:`
- `HKEY_LOCAL_MACHINE / HKLM:`
- `HKEY_CURRENT_CONFIG / HKCC:`
- `HKEY_CURRENT_USER / HKCU:`

Tipos admitidos

- `BINARY`
- `DWORD`
- `QWORD`
- `SZ`
- `EXPAND_SZ`

- MULTI_SZ

Ejemplo de entrada: cómo establecer los valores de las claves de registro

```
- name: SetRegistryKeyValues
  action: SetRegistry
  maxAttempts: 3
  inputs:
    - path: HKLM:\SOFTWARE\MySoftWare
      name: MyName
      value: FirstVersionSoftware
      type: SZ
    - path: HKEY_CURRENT_USER\Software\Test
      name: Version
      value: 1.1
      type: DWORD
```

Salida

Ninguna.

UpdateOS (Linux, Windows)

El módulo de acción UpdateOS añade soporte para instalar actualizaciones de Windows y Linux. Instala todas las actualizaciones disponibles de forma predeterminada. Como alternativa, puede configurar una lista de una o más actualizaciones específicas para que las instale el módulo de acción. También puede especificar las actualizaciones para excluirlas de la instalación.

Si se proporcionan listas de "incluir" y "excluir", la lista de actualizaciones resultante solo podrá incluir las incluidas en la lista de "inclusión" que no estén incluidas en la lista de "excluidas".

> **Note**
>
> UpdateOS no es compatible con Amazon Linux 2023 (AL2023). Le recomendamos que actualice la AMI base a la nueva versión que viene con cada versión. Para ver otras alternativas, consulte Controlar las actualizaciones recibidas de las versiones principales y secundarias en la Guía del usuario de Amazon Linux 2023.

- Windows. Las actualizaciones se instalan desde la fuente de actualización configurada en la máquina de destino.

- Linux. La aplicación busca el administrador de paquetes compatible en la plataforma Linux y utiliza un administrador de paquetes `yum` o `apt-get`. Si ninguna de las opciones es compatible, se devuelve un error. Debe tener permisos `sudo` para ejecutar el módulo de acción UpdateOS. Si no tiene permisos `sudo` se devuelve un `error.Input`.

Input

Nombre de la clave	Descripción	Tipo	Obligatorio/a
`include`	Para Windows puede especificar lo siguiente: • Uno o más artículos de Microsoft Knowledge Base (KB) IDs para incluirlos en la lista de actualizaciones que se pueden instalar. Los formatos válidos son `KB1234567` o `1234567`. • Un nombre de actualización con un valor comodín (`*`). Los formatos válidos son `Security*` o `*Security*` . Para Linux, puede especificar uno o	Lista de cadenas	No

Nombre de la clave	Descripción	Tipo	Obligatorio/a
	más paquetes para incluirlos en la lista de actualizaciones para su instalación.		

Nombre de la clave	Descripción	Tipo	Obligatorio/a
`exclude`	Para Windows puede especificar lo siguiente: • Uno o más artículos de Microsoft Knowledge Base (KB) IDs para incluirlos en la lista de actualizaciones que se van a excluir de la instalación. Los formatos válidos son `KB1234567` o `1234567`. • Un nombre de actualización con un valor comodín (`*`). Los formatos válidos son `Security*` o `*Security*` . Para Linux, puede especificar uno o más paquetes para incluirlos en la lista de actualizaciones para su instalación.	Lista de cadenas	No

Ejemplo de entrada: cómo añadir soporte para instalar actualizaciones de Linux

```
- name: UpdateMyLinux
  action: UpdateOS
  onFailure: Abort
  maxAttempts: 3
  inputs:
    exclude:
      - ec2-hibinit-agent
```

Ejemplo de entrada: cómo añadir soporte para instalar actualizaciones de Windows

```
- name: UpdateWindowsOperatingSystem
  action: UpdateOS
  onFailure: Abort
  maxAttempts: 3
  inputs:
    include:
      - KB1234567
      - '*Security*'
```

Salida

Ninguna.

Configurar la entrada para el comando TOE de AWS run

Para agilizar la introducción de los comandos desde la línea de TOE de AWS run comandos, puede incluir los ajustes de los parámetros y las opciones de los comandos en un archivo de configuración de entrada en formato JSON con una extensión de `.json` archivo. TOE de AWS puede leer el archivo desde una de las siguientes ubicaciones:

- Una ruta de archivo local (*`./config.json`*).
- Un bucket de S3 (*`s3://<bucket-path>/<bucket-name>/config.json`*).

Cuando introduce el comando run, puede especificar el archivo de configuración de entrada mediante el parámetro --config. Por ejemplo:

```
awstoe run --config <file-path>/config.json
```

Archivo de configuración de entrada

El archivo JSON de configuración de entrada incluye pares clave-valor para todos los ajustes que puede proporcionar directamente a través de los parámetros y las opciones de los comandos run. Si especifica un ajuste tanto en el archivo de configuración de entrada como en el comando run, como parámetro u opción, se aplicarán las siguientes reglas de prioridad:

Reglas de prioridad

1. Un ajuste que se proporciona directamente al run comando en el AWS CLI, mediante un parámetro o una opción, anula cualquier valor que esté definido en el archivo de configuración de entrada para el mismo ajuste.
2. Un parámetro del archivo de configuración de entrada anula el valor predeterminado de un componente.
3. Si no se pasa ninguna otra configuración al documento del componente, se puede aplicar un valor por defecto, si lo hay.

Hay dos excepciones a esta regla: los documentos y los parámetros. Estos ajustes funcionan de forma diferente en la configuración de entrada y como parámetros de comando. Si utiliza el archivo de configuración de entrada, no debe especificar estos parámetros directamente en el comando run. Si lo hace, se generará un error.

Ajustes de los componentes

El archivo de configuración de entrada contiene los siguientes ajustes. Para simplificar el archivo, puede omitir cualquier configuración opcional que no sea necesaria. Todos los ajustes son opcionales, a menos que se indique lo contrario.

- cwIgnoreFailures(Booleano): ignore los errores de registro de los registros. CloudWatch
- cwLogGroup(String): el `LogGroup` nombre de los CloudWatch registros.
- cwLogRegion(Cadena): la AWS región que se aplica a los CloudWatch registros.
- cwLogStream(Cadena): el `LogStream` nombre de los CloudWatch registros, que indica TOE de AWS dónde se debe transmitir el `console.log` archivo.
- DocumentS3 BucketOwner (String): el ID de cuenta del propietario del bucket para los documentos basados en URI de S3.

- documentos (conjunto de objetos, obligatorio): conjunto de objetos JSON que representa los documentos del componente YAML que ejecuta el TOE de AWS run comando. Debe especificarse al menos un documento componente.

  Cada objeto consiste en los siguientes campos:

  - Ruta (cadena, obligatoria): la ubicación del archivo del documento del componente YAML. Este debe ser uno de los siguientes:
    - Una ruta de archivo local (*./component-doc-example.yaml*).
    - Un URI de S3 (s3://*bucket/key*).
    - Un componente de Image Builder compiló la versión ARN (arn:aws:imagebuilder:us-west-:component/ /2021.12.02/1). *2:123456789012 my-example-component*
  - Parámetros (matriz de objetos): matriz de objetos de pares clave-valor, cada uno de los cuales representa un parámetro específico del componente que el comando run transfiere al ejecutar el documento del componente. Los parámetros son opcionales para los componentes. El documento componente puede o no tener parámetros definidos.

    Cada objeto consiste en los siguientes campos:

    - Nombre (cadena, obligatorio): el nombre del parámetro del componente.
    - Valor (cadena, obligatorio): el valor que se va a transferir al documento del componente para el parámetro nombrado.

    Para obtener más información sobre los parámetros de los componentes, consulte la sección Parámetros de la página Uso de variables en su documento de componentes personalizados.
- executonID (cadena): este es el identificador único que se aplica a la ejecución del comando run actual. Este identificador se incluye en los nombres de los archivos de salida y de registro para identificarlos de forma exclusiva y vincularlos a la ejecución del comando actual. Si se omite TOE de AWS esta configuración, genera un GUID.
- LogDirectory (String): el directorio de destino donde se TOE de AWS almacenan todos los archivos de registro de la ejecución de este comando. De forma predeterminada, este directorio está ubicado en el siguiente directorio principal: TOE_<DATETIME>_<EXECUTIONID>. Si no especifica el directorio de registro, TOE de AWS utiliza el directorio de trabajo actual (.).
- LogS3 BucketName (String): si los registros de los componentes están almacenados en Amazon S3 (recomendado), TOE de AWS carga los registros de las aplicaciones de los componentes en el bucket de S3 mencionado en este parámetro.

- LogS3 BucketOwner (String): si los registros de los componentes se almacenan en Amazon S3 (recomendado), este es el ID de cuenta propietario del depósito donde se TOE de AWS escriben los archivos de registro.
- LogS3 KeyPrefix (String): si los registros de los componentes se almacenan en Amazon S3 (recomendado), este es el prefijo de clave de objeto S3 para la ubicación del registro en el bucket.
- Parámetros (matriz de objetos): una matriz de objetos de pares clave-valor que representan parámetros que se aplican globalmente a todos los componentes que se incluyen en la ejecución del comando run actual.
  - Nombre (cadena, obligatorio): el nombre del parámetro global.
  - Valor (cadena, obligatorio): el valor que se va a transferir a todos los documentos de componentes del parámetro nombrado.
- Fases (cadena): una lista separada por comas que especifica qué fases se van a ejecutar desde los documentos de componentes de YAML. Si un documento de componentes incluye fases adicionales, no se ejecutarán.
- StateDirectory (cadena): la ruta del archivo donde se almacenan los archivos de seguimiento de estado.
- Huella (booleano): permite el registro detallado en la consola.

Ejemplos

El siguiente ejemplo muestra un archivo de configuración de entrada que ejecuta las fases `build` y `test` de dos documentos componentes: `sampledoc.yaml` y `conversation-intro.yaml`. Cada documento componente tiene un parámetro que se aplica solo a sí mismo y ambos utilizan un parámetro compartido. El parámetro `project` se aplica a ambos documentos componentes.

```
{
   "documents": [
     {
       "path": "<file path>/awstoe/sampledoc.yaml>",
       "parameters": [
         {
           "name": "dayofweek",
           "value": "Monday"
         }
       ]
     },
     {
```

```
      "path": "<file path>/awstoe/conversation-intro.yaml>",
      "parameters": [
        {
          "name": "greeting",
          "value": "Hello, HAL."
        }
      ]
    }
  ],
  "phases": "build,test",
  "parameters": [
    {
      "name": "project",
      "value": "examples"
    }
  ],
  "cwLogGroup": "<log_group_name>",
  "cwLogStream": "<log_stream_name>",
  "documentS3BucketOwner": "<owner_aws_account_number>",
  "executionId": "<id_number>",
  "logDirectory": "<local_directory_path>",
  "logS3BucketName": "<bucket_name_for_log_files>",
  "logS3KeyPrefix": "<key_prefix_for_log_files>",
  "logS3BucketOwner": "<owner_aws_account_number>"
}
```

Recursos de imágenes resultantes de Generador de Imágenes

Después de crear recursos de imagen para AMI o imágenes de contenedor con Generador de Imágenes, puede administrarlos usando la consola Generador de Imágenes, mediante la API de Generador de Imágenes o con los comandos imagebuilder en la AWS CLI.

> **Tip**
>
> Cuando tenga muchos recursos del mismo tipo, el etiquetado puede ayudarle a identificar un recurso específico en función de las etiquetas que le haya asignado. Para obtener más información sobre cómo etiquetar los recursos mediante los comandos de Image Builder de AWS CLI, consulte la Etiquetar recursos sección de esta guía.

En esta sección se explica cómo enumerar, ver y crear imágenes. Para obtener información sobre los flujos de trabajo de imágenes y cómo administrarlos, consulte Administración de flujos de trabajo de creación y prueba para imágenes de Generador de imágenes.

Contenido

- Enumeración de imágenes y versiones de compilación
- Ver detalles de recurso de imagen
- Creación de imágenes personalizadas con Generador de imágenes
- Importación y exportación de imágenes de máquinas virtuales con Generador de imágenes
- Importe imágenes de disco ISO de Windows verificadas con Image Builder
- Administración de los resultados de seguridad para las imágenes de Generador de Imágenes
- Limpieza de los recursos de Generador de imágenes

Enumeración de imágenes y versiones de compilación

En la página Imágenes de la consola de Generador de imágenes, puede ver listas de todos los recursos de imágenes de Generador de imágenes que le pertenecen, que han compartido con usted y a los que tiene acceso. Los resultados de la lista incluyen algunos detalles clave sobre esos recursos.

También puede ver todas las imágenes de la cuenta que tienen pendientes acciones de flujo de trabajo.

Contenido

- Enumeración de imágenes
- Enumeración de imágenes en espera de acción
- Enumeración de las versiones de compilación de imágenes

Enumeración de imágenes

En esta sección se describen las distintas formas de enumerar la información sobre las imágenes.

Puede usar uno de los siguientes métodos para enumerar los recursos de imágenes de Generador de Imágenes a los que tiene acceso. Para ver la acción de la API, consulte ListImagesla Referencia de la API de EC2 Image Builder. Para la solicitud de SDK asociada, consulte el enlace Véase también en la misma página.

Contenido

- Enumeración de las imágenes en la consola
- Enumere las imágenes con comandos AWS CLI

Enumeración de las imágenes en la consola

Para abrir la página de lista Imágenes en la consola, sigue estos pasos:

1. Abra la consola EC2 de Image Builder en https://console.aws.amazon.com/imagebuilder/.
2. Elija Imágenes en el panel de navegación.

La página Imágenes de la consola se divide en pestañas, según la propiedad de la imagen o las acciones pendientes del flujo de trabajo. En esta sección se describen las tres primeras pestañas en las que se muestran las imágenes de su propiedad o a las que tiene acceso.

Pestaña de la consola: De mi propiedad

En la pestaña De mi propiedad, puede utilizar los siguientes filtros para optimizar los resultados de la lista de imágenes.

- En la barra de búsqueda, puede buscar la totalidad o parte del nombre.
- Puede filtrar las imágenes en función de la plataforma del sistema operativo (Windows, Linux o macOS).
- Puede filtrar las imágenes en función del tipo de salida que producen (imagen de AMI o de contenedor).
- Puede utilizar la fuente del filtro para buscar imágenes importadas desde una máquina virtual (VMIE) o desde una imagen de disco ISO.

Siguiendo los controles de filtro, en la pestaña De mi propiedad se muestra una lista de las imágenes de Generador de imágenes que ha creado, con los siguientes detalles de los recursos de la lista:

Nombre y versión

Los nombres de los recursos de imagen de Generador de imágenes comienzan con el nombre de la receta y la versión a partir de la cual se crearon. Seleccione el enlace para ver todas las versiones de creación de imágenes relacionadas.

Tipo

Tipo de la imagen de salida que Generador de imágenes crea para este recurso de imagen (una imagen de AMI o de contenedor).

Plataforma

Plataforma del sistema operativo del recurso de imagen, por ejemplo, "Linux", "Windows" o "macOS".

Fuente de imagen

Origen de la imagen base que Generador de imágenes utilizó para crear este recurso de imagen. Se utiliza principalmente para filtrar los resultados de las imágenes importadas desde una máquina virtual (VMIE).

Hora de creación

Fecha y hora en que Generador de imágenes creó la versión actual del recurso de imagen.

ARN

Nombre de recurso de Amazon (ARN) de la versión actual del recurso de imagen.

Pestaña de la consola: Compartido conmigo

En la pestaña Compartido conmigo, puede utilizar los siguientes filtros para optimizar los resultados de la lista de imágenes.

- En la barra de búsqueda, puede buscar la totalidad o parte del nombre.
- Puede filtrar las imágenes en función de la plataforma del sistema operativo (Windows, Linux o macOS).
- Puede filtrar las imágenes en función del tipo de salida que producen (imagen de AMI o de contenedor).
- Puede utilizar la fuente del filtro para buscar imágenes importadas de una máquina virtual (VMIE) o de una imagen de disco ISO.

Siguiendo los controles de filtro, en la pestaña Compartido conmigo se muestra una lista de las imágenes de Generador de imágenes que se han compartido con usted, con los siguientes detalles de los recursos de la lista:

Nombre de la imagen

Nombre del recurso de imagen que se compartió con usted. Para utilizar una imagen compartida en una receta, seleccione la opción Seleccionar imágenes administradas y cambie el Origen de la imagen a Imágenes compartidas conmigo.

Tipo

Tipo de la imagen de salida que Generador de imágenes crea para este recurso de imagen (una imagen de AMI o de contenedor).

Versión

La versión de la plataforma del sistema operativo para el recurso de imagen, normalmente un campo numérico con el siguiente formato:<major>. <minor>. <patch>.

Fuente de imagen

Origen de la imagen base que Generador de imágenes utilizó para crear este recurso de imagen, si corresponde. Se utiliza principalmente para filtrar los resultados de las imágenes importadas desde una máquina virtual (VMIE).

Plataforma

Plataforma del sistema operativo del recurso de imagen, por ejemplo, "Linux", "Windows" o "macOS".

Hora de creación

Fecha y hora en que Generador de imágenes creó la versión del recurso de imagen que se compartió con usted.

Propietario

Propietario del recurso de imagen compartido.

ARN

Nombre de recurso de Amazon (ARN) de la versión del recurso de imagen que se compartió con usted.

Pestaña de la consola: Administrado por Amazon

En la pestaña Administrado por Amazon, puede utilizar los siguientes filtros para optimizar los resultados de la lista de imágenes.

- En la barra de búsqueda, puede buscar la totalidad o parte del nombre.
- Puede filtrar las imágenes en función de la plataforma del sistema operativo (Windows, Linux o macOS).
- Puede filtrar las imágenes en función del tipo de salida que producen (imagen de AMI o de contenedor).
- Puede utilizar la fuente del filtro para buscar imágenes importadas de una máquina virtual (VMIE) o de una imagen de disco ISO.

Siguiendo los controles de filtro, en la pestaña Administrado por Amazon se muestra una lista de imágenes de Generador de imágenes administradas por Amazon que puede utilizar como imágenes base para las recetas. En Generador de imágenes se muestran los siguientes detalles de los recursos enumerados:

Nombre de la imagen

Nombre de la imagen administrada. Al crear una receta, el valor predeterminado de la imagen base es Inicio rápido (administrado por Amazon). Las imágenes que aparecen en esta pestaña

rellenan la lista Nombre de la imagen, asociada a la plataforma del sistema operativo que elija como imagen base al crear una receta.

Tipo

Tipo de la imagen de salida que Generador de imágenes crea para este recurso de imagen (una imagen de AMI o de contenedor).

Versión

La versión de la plataforma del sistema operativo para el recurso de imagen, normalmente un campo numérico con el siguiente formato:<major>. <minor>. <patch>.

Plataforma

Plataforma del sistema operativo del recurso de imagen, por ejemplo, "Linux", "Windows" o "macOS".

Hora de creación

Fecha y hora en que Generador de imágenes creó la versión del recurso de imagen que se compartió con usted.

Propietario

Las imágenes administradas son propiedad de Amazon.

ARN

Nombre de recurso de Amazon (ARN) de la versión del recurso de imagen que se compartió con usted.

Enumere las imágenes con comandos AWS CLI

Al ejecutar el list-images comando en el AWS CLI, puede obtener una lista de imágenes de su propiedad o a las que tiene acceso.

En el siguiente ejemplo de comando se muestra cómo utilizar el comando list-images sin filtros para enumerar todos los recursos de imágenes de Generador de imágenes de su propiedad.

Ejemplo: enumerar todas las imágenes

```
aws imagebuilder list-images
```

Salida:

```
{
 "requestId": "1abcd234-e567-8fa9-0123-4567b890cd12",
 "imageVersionList": [
  {
    "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image/image-recipe-name/1.0.0",
    "name": "image-recipe-name",
    "type": "AMI",
    "version": "1.0.0",
    "platform": "Linux",
    "owner": "123456789012",
    "dateCreated": "2022-04-28T01:38:23.286Z"
  },
  {
    "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image/image-recipe-win/1.0.1",
    "name": "image-recipe-win",
    "type": "AMI",
    "version": "1.0.1",
    "platform": "Windows",
    "owner": "123456789012",
    "dateCreated": "2022-04-28T01:38:23.286Z"
  },
  {
    "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image/image-recipe-
macos/1.1.1",
    "name": "image-recipe-macos",
    "type": "AMI",
    "version": "1.1.1",
    "platform": "macOS",
    "owner": "123456789012",
    "dateCreated": "2022-04-28T01:38:23.286Z"
  }
 ]
}
```


Al ejecutar el comando list-images, puede aplicar filtros para agilizar los resultados, como se muestra en el siguiente ejemplo. Para obtener más información sobre cómo filtrar los resultados, consulte el comando list-images en la referencia de comandos de la AWS CLI .

Ejemplo: filtro para imágenes de Linux

```
aws imagebuilder list-images --filters name="platform",values="Linux"
```

Salida:

```
{
 "requestId": "1abcd234-e567-8fa9-0123-4567b890cd12",
 "imageVersionList": [
  {
    "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image/image-recipe-name/1.0.0",
    "name": "image-recipe-name",
    "type": "AMI",
    "version": "1.0.0",
    "platform": "Linux",
    "owner": "123456789012",
    "dateCreated": "2022-04-28T01:38:23.286Z"
  }
 ]
}
```

Enumeración de imágenes en espera de acción

Cuando se utiliza la acción de paso `WaitForAction` en el flujo de trabajo de imágenes, se pausa el flujo de trabajo hasta que le envíe una señal para reanudar el procesamiento o se produzca un error en el flujo de trabajo. Puede utilizar esta acción de paso si tiene un proceso externo que debe ejecutarse antes de continuar. A continuación, puede utilizar `SendWorkflowStepAction` para enviar una señal al paso en pausa para `RESUME` o `STOP`. También puede detener o reanudar el flujo de trabajo desde la consola.

En las siguientes pestañas se muestra cómo obtener una lista de todos los recursos de imagen de su cuenta con los pasos del flujo de trabajo que actualmente están en pausa en espera de una señal para reanudarse o detenerse. Las pestañas muestran los pasos de la consola y el AWS CLI comando.

También puede utilizar la API o un SDK para obtener una lista de los pasos del flujo de trabajo que están pendientes de acción. Para ver la acción de la API, consulte ListWaitingWorkflowStepsla Referencia de la API de EC2 Image Builder. Para la solicitud de SDK asociada, consulte el enlace Véase también en la misma página.

Console

Para acceder a la pestaña Esperando acción de la consola, siga estos pasos:

1. Abra la consola EC2 de Image Builder en https://console.aws.amazon.com/imagebuilder/.

2. Elija Imágenes en el panel de navegación. Se abrirá la página de lista Imágenes.
3. Seleccione la pestaña Esperando acción en la página de lista.
4. (Opcional) Para detener o reanudar un paso, active la casilla situada junto al nombre y, a continuación, elija Detener paso o Reanudar paso. Puede seleccionar más de una casilla para realizar la misma acción en todos los pasos seleccionados.

Detalles de los pasos pendientes del flujo de trabajo

Los detalles del flujo de trabajo del paso pendiente incluyen lo siguiente:

- Nombre de la imagen: nombre del recurso de imagen que tiene el paso pendiente. Puede seleccionar el enlace del nombre para mostrar la página de detalles de esa imagen.
- Nombre del paso pendiente: nombre del paso del flujo de trabajo que está esperando la acción.
- ID de ejecución del paso: identifica de forma exclusiva la instancia de tiempo de ejecución del paso del flujo de trabajo. Puede seleccionar el ID vinculado para mostrar los detalles del tiempo de ejecución del paso.
- Inicio del paso: marca de tiempo del inicio de la instancia de tiempo de ejecución del paso en el flujo de trabajo.
- ARN de flujo de trabajo: nombre de recurso de Amazon (ARN) del flujo de trabajo con el paso pendiente.
- Acciones: acción del paso que está en estado de espera.

AWS CLI

Cuando ejecutes el list-waiting-workflow-steps comando en AWS CLI, obtendrás una lista de todas las imágenes de tu cuenta que incluyen pasos de flujo de trabajo que están pendientes de acción antes de completar el proceso de creación de la imagen.

En el siguiente ejemplo de comando, se muestra cómo utilizar el comando list-waiting-workflow-steps para enumerar todas las imágenes de su cuenta con pasos del flujo de trabajo que están en espera de una acción.

Ejemplo: enumeración de imágenes de la cuenta con los pasos del flujo de trabajo en espera

```
aws imagebuilder list-waiting-workflow-steps
```

Salida:

En la salida de este ejemplo se muestra una imagen de la cuenta con un paso en espera de una acción.

```
{
    "steps": [
        {
            "imageBuildVersionArn": "arn:aws:imagebuilder:us-
west-2:111122223333:image/example-image/1.0.0/8",
            "name": "WaitForAction",
            "workflowExecutionId": "wf-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "stepExecutionId": "step-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
            "workflowBuildVersionArn": "arn:aws:imagebuilder:us-
west-2:111122223333:workflow/test/wait-for-action/1.0.0/1",
            "startTime": "2023-11-21T23:21:23.609Z",
            "action": "WaitForAction"
        }
    ]
}
```

Enumeración de las versiones de compilación de imágenes

En la página Versiones de compilación de imágenes de la consola de Generador de Imágenes, puede ver una lista de versiones de compilación y detalles adicionales de un recurso de imagen de su propiedad. También puede usar comandos o acciones con la API Image Builder o AWS CLI para enumerar las versiones de creación de imágenes. SDKs

Puede usar uno de los siguientes métodos para enumerar las versiones de compilación de imágenes para los recursos de imágenes de su propiedad. Para ver la acción de la API, consulte ListImageBuildVersionsla Referencia de la API de EC2 Image Builder. Para la solicitud de SDK asociada, consulte el enlace Véase también en la misma página.

Console

Detalles de la versión

Los detalles de la página Versiones de compilación de imágenes en la consola de Generador de Imágenes incluyen lo siguiente:

- versión: la versión de compilación del recurso de imagen. En la consola de Generador de Imágenes, la versión se enlaza con una página de detalles de la imagen.

- Tipo: el tipo de salida que Generador de Imágenes distribuyó al crear este recurso de imagen (una AMI o una imagen de contenedor).
- Fecha de creación: fecha y hora en que Generador de Imágenes creó la versión de compilación de imágenes.
- Estado de la imagen: el estado actual de la versión de compilación de la imagen. El estado puede estar relacionado con la compilación o disposición de la imagen. Por ejemplo, durante el proceso de compilación, es posible que vea un estado `Building` o `Distributing`. Para la disposición de la imagen, es posible que vea un estado `Deprecated` o `Deleted`.
- Motivo del error: el motivo del estado de la imagen. La consola de Generador de Imágenes solo muestra el motivo por el que se produce un error en la compilación (“Image status equals `Failed`“).
- Resultados de seguridad: los resultados agregados del escaneo de imágenes para la versión de compilación de imágenes a la que se hace referencia.
- ARN: el nombre de recurso de Amazon (ARN) de la versión a la que se hace referencia del recurso de imagen.
- Flujo de registro: enlace a los detalles del flujo de registro de la versión de compilación de la imagen a la que se hace referencia.

Enumeración de versiones

Para enumerar las versiones de compilación de imágenes en la consola de Generador de Imágenes, lleve a cabo los siguientes pasos:

1. Abra la consola EC2 de Image Builder en https://console.aws.amazon.com/imagebuilder/.
2. Elija Imágenes en el panel de navegación. De forma predeterminada, la lista de imágenes muestra la versión actual de cada una de las imágenes que posee.
3. Para ver una lista de todas las versiones de una imagen, seleccione el enlace de la versión actual. El enlace abre la página Versiones de compilación de imágenes, que muestra todas las versiones de compilación de una imagen específica.

AWS CLI

Al ejecutar el list-image-build-versions comando en AWS CLI, obtendrá una lista completa de las versiones de compilación del recurso de imagen especificado. Debe ser el propietario de la imagen para ejecutar este comando.

El siguiente ejemplo de comando muestra cómo usar el comando list-image-build-versions para enumerar todas las versiones de compilación de la imagen especificada.

Ejemplo: enumerar las versiones de compilación de una imagen específica.

```
aws imagebuilder list-image-build-versions --image-version-arn
 arn:aws:imagebuilder:us-west-2:123456789012:image/image-recipe-name/1.0.0
```

Salida:

El resultado de este ejemplo incluye dos versiones de compilación para la receta de imagen especificada.


```
{
  "requestId": "12f3e45d-67cb-8901-af23-45ed678c9b01",
  "imageSummaryList": [
    {
    "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image/image-recipe-
name/1.0.0/2",
    "name": "image-recipe-name",
    "type": "AMI",
    "version": "1.0.0/2",
    "platform": "Linux",
    "osVersion": "Amazon Linux 2",
    "state": {
      "status": "AVAILABLE"
    },
    "owner": "123456789012",
    "dateCreated": "2023-03-10T01:04:40.609Z",
    "outputResources": {
      "amis": [
        {
        "region": "us-west-2",
        "image": "ami-012b3456789012c3d",
        "name": "image-recipe-name 2023-03-10T01-05-12.541Z",
        "description": "First verison of image-recipe-name",
        "accountId": "123456789012"
        }
      ]
    },
    "tags": {}
    },
    {
```

```
    "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image/image-recipe-
name/1.0.0/1",
    "name": "image-recipe-name",
    "type": "AMI",
    "version": "1.0.0/1",
    "platform": "Linux",
    "osVersion": "Amazon Linux 2",
    "state": {
      "status": "AVAILABLE"
    },
    "owner": "123456789012",
    "dateCreated": "2023-03-10T00:07:16.384Z",
    "outputResources": {
      "amis": [
        {
        "region": "us-west-2",
        "image": "ami-0d1e23456789f0a12",
        "name": "image-recipe-name 2023-03-10T00-07-18.146132Z",
        "description": "First verison of image-recipe-name",
        "accountId": "123456789012"
        }
      ]
    },
    "tags": {}
    }
  ]
}
```


> **Note**
>
> El resultado del comando list-image-build-versions no incluye los resultados de seguridad ni los flujos de registro en este momento.

Ver detalles de recurso de imagen

En la página de detalles de la imagen de la consola de Generador de Imágenes, puede ver los detalles de un recurso de imagen específico de su propiedad. También puede usar comandos o acciones con la API Image Builder o AWS CLI para obtener detalles de la imagen. SDKs

Para obtener más información sobre los recursos que otra Cuenta de AWS persona compartió contigo a través de un recurso compartido AWS Resource Access Manager (AWS RAM), consulta Acceder a AWS los recursos compartidos contigo en la Guía del AWS RAM usuario.

Contenido

- Visualización de detalles de la imagen en la consola de Generador de Imágenes
- Obtenga los detalles de la política de imágenes en el AWS CLI

Visualización de detalles de la imagen en la consola de Generador de Imágenes

La página de detalles de la imagen de la consola de Generador de Imágenes incluye una sección de resumen con información adicional agrupada en pestañas. El encabezado de la página es el nombre y la versión de compilación de la receta que creó la imagen. Si una pestaña no se aplica a tu imagen, estará inactiva y no mostrará datos.

Secciones y pestañas de detalles de la consola

- Sección de resumen
- Pestaña de recursos de salida
- Pestaña de configuración de infraestructura
- Pestaña de configuración de distribución
- Pestaña de flujo de trabajo
- Pestaña de resultados de seguridad
- Pestaña Etiquetas

Sección de resumen

La sección de resumen abarca el ancho de la página e incluye los siguientes detalles. Estos detalles siempre se muestran.

Receta

> El nombre y la versión de la receta que no incluye la versión de compilación. Por ejemplo, si la versión de compilación es `sample-linux-recipe | 1.0.1/2`, la receta es `sample-linux-recipe | 1.0.1` y la versión de compilación es 2.

Date created (Fecha de creación)

Fecha y hora en que Generador de Imágenes creó la versión de compilación de imágenes.

Estado de la imagen

El estado actual de la versión de compilación de la imagen. El estado puede estar relacionado con la compilación o disposición de la imagen. Por ejemplo, durante el proceso de compilación, es posible que vea un estado `Building` o `Distributing`. Para la disposición de la imagen, es posible que vea un estado `Deprecated` o `Deleted`.

Motivo del error

El motivo del estado de la imagen. La consola de Generador de Imágenes solo muestra el motivo por el que se produce un error en la compilación ("Image status equals `Failed`").

Pestaña de recursos de salida

La pestaña Recursos de salida muestra los detalles de salida y distribución del recurso de imagen que se muestra actualmente. La información que muestra Generador de Imágenes depende del tipo de receta que la canalización utilizó para crear la imagen, de la siguiente manera.

Receta de imagen

- Región: la región de distribución de la imagen de máquina de Amazon (AMI) de salida que se especifica en la columna Imagen.
- Imagen: el ID de la AMI que Generador de Imágenes distribuyó al destino. Este ID está vinculado a la página Amazon Machine Images (AMIs) de la EC2 consola de Amazon.

> **Note**
>
> Generador de Imágenes crea la AMI después de crear el recurso de imagen de salida y antes de distribuir la AMI al destino.

- Nombre: el nombre de la AMI que Generador de Imágenes distribuyó al destino.
- Descripción: la descripción opcional de la receta de imagen que la canalización utilizó para crear el recurso de imagen de salida.
- Cuenta: la propietaria del recurso de imagen Image Builder Cuenta de AWS que se muestra actualmente.

Receta de contenedor

Generador de Imágenes muestra los siguientes detalles de la salida creada a partir de una receta de contenedor.

- Región: la región de distribución de la imagen de contenedor que se especifica en la columna URI de imagen.
- URI de imagen: URI de imagen del contenedor de salida que Generador de Imágenes distribuyó al repositorio de ECR de la región de destino.

> **Note**
>
> Generador de Imágenes muestra una fila por destino. La imagen de salida siempre tiene al menos una entrada para distribuirla en la cuenta que creó la imagen. Los destinos adicionales pueden incluir distribuciones entre regiones Cuentas de AWS, o AWS Organizations. Para obtener más información, consulte Administrar los ajustes de la distribución de Generador de imágenes.

Pestaña de configuración de infraestructura

La pestaña Configuración de infraestructura muestra los ajustes de EC2 infraestructura de Amazon que Image Builder utilizó para crear y probar la imagen que se muestra actualmente. Generador de Imágenes siempre muestra el nombre del recurso de configuración de la infraestructura (nombre de configuración) y su nombre de recurso de Amazon (ARN). Si la configuración de la infraestructura establece los valores, los detalles adicionales de la infraestructura pueden incluir lo siguiente:

- Tipos de instancias
- Un perfil de instancia
- Infraestructura de red
- Configuración de grupo de seguridad
- Ubicación de Amazon S3 donde Generador de Imágenes almacena los registros de las aplicaciones
- Un par de EC2 claves de Amazon para solucionar problemas
- Un tema de Amazon SNS para las notificaciones de eventos

Para obtener más información, consulte Administre la configuración de la infraestructura de Generador de imágenes.

Pestaña de configuración de distribución

La pestaña Ajustes de distribución muestra los ajustes que Generador de Imágenes utilizó para distribuir las imágenes de salida. Generador de Imágenes siempre muestra el nombre del recurso de configuración de la distribución (nombre de configuración) y su nombre de recurso de Amazon (ARN). Los detalles de distribución adicionales dependen del tipo de receta que la canalización de Generador de Imágenes utilizó para compilar la imagen, de la siguiente manera:

Receta de imagen

Si su recurso de configuración de distribución establece los valores, los detalles de distribución adicionales pueden incluir lo siguiente:

- Región: la región de distribución de la imagen de máquina de Amazon (AMI) de salida.
- Nombre de la AMI de salida: el nombre de la AMI que Generador de Imágenes distribuyó al destino.
- Cifrado (clave KMS): si está configurado, el AWS KMS key que Generador de Imágenes utiliza para cifrar la imagen y distribuirla en la región de destino.
- Cuentas de destino para la distribución: si has configurado la distribución entre cuentas, en esta columna se muestra una lista separada por comas con las Cuentas de AWS que compartir la imagen de salida en la región de destino.
- Directores con permisos compartidos: una lista separada por comas de AWS los directores que tienen permiso para lanzar tu imagen, por ejemplo, grupos Cuentas de AWS o unidades organizativas (). AWS Organizations OUs

> **Note**
>
> Si autorizas a otros directores a lanzar tu imagen, la imagen seguirá siendo tuya. AWS factura a tu cuenta todas las instancias que Amazon EC2 lance a partir de tu imagen.

- Target cuenta con una configuración de lanzamiento más rápida: ahí es Cuentas de AWS donde EC2 Fast Launch distribuye las instantáneas previamente aprovisionadas para el lanzamiento.
- Configuraciones de licencia asociadas: la configuración de licencia de License Manager ARNs que está asociada a la AMI en la región especificada.

- Configuración de la plantilla de lanzamiento: identifica una plantilla de EC2 lanzamiento de Amazon para utilizarla en una cuenta específica.
- Establecer la versión predeterminada de la plantilla de lanzamiento: establece la plantilla de EC2 lanzamiento de Amazon especificada como la plantilla de lanzamiento predeterminada para la especificada Cuenta de AWS.

Receta de contenedor

Las distribuciones de contenedores siempre incluyen los siguientes detalles:

- Región: la región de distribución de la imagen de contenedor especificada en la columna URI de imagen.
- URI de imagen: URI de imagen del contenedor de salida que Generador de Imágenes distribuyó al repositorio de Amazon ECR de la región de destino.

> **Note**
>
> Generador de Imágenes muestra una fila por destino. La imagen de salida siempre tiene al menos una entrada para distribuirla en la cuenta que creó la imagen. Los destinos adicionales pueden incluir distribuciones en todas las regiones Cuentas de AWS, o AWS Organizations. Para obtener más información, consulte Administrar los ajustes de la distribución de Generador de imágenes.

Pestaña de flujo de trabajo

Los flujos de trabajo definen la secuencia de pasos que Generador de Imágenes realiza al crear una nueva imagen. Todas las imágenes tienen flujos de trabajo de creación, prueba y distribución. La pestaña Flujo de trabajo muestra los flujos de trabajo aplicables que Generador de Imágenes ejecutó para la imagen.

Cómo filtrar tipos de flujo de trabajo

Image Builder muestra inicialmente el resumen del flujo de trabajo de creación o importación y los pasos del flujo de trabajo de forma predeterminada. Sin embargo, el filtro de Flujo de trabajo muestra todos los flujos de trabajo que están en progreso o completados para su imagen. Para ver un flujo de trabajo diferente, selecciónelo de la lista.

Los flujos de trabajo de imágenes que producen resultados de AMI pueden tener flujos de trabajo de creación, importación o prueba. Los flujos de trabajo de contenedores que producen una salida de contenedor pueden tener flujos de trabajo de creación, prueba o distribución.

> **Note**
>
> Si el flujo de trabajo aún no se ha iniciado, no aparecerá en la lista. Por ejemplo, si la creación de imágenes que tiene configurados flujos de trabajo de compilación y prueba acaba de empezar, el flujo de trabajo de creación es el único tipo de flujo de trabajo que aparece en la lista. Cuando comienza el flujo de trabajo de prueba, Image Builder lo añade a la lista.

Después del filtro de Flujo de trabajo, el flujo de trabajo seleccionado muestra un resumen del tiempo de ejecución que incluye los siguientes detalles para cada tipo de flujo de trabajo:

Estado del flujo de trabajo

El estado de tiempo de ejecución actual de este flujo de trabajo. Los valores pueden incluir lo siguiente:

- Pending (Pendiente)
- Skipped
- Running
- Completado
- Con error
- R ollback-in-progress
- Rollback-completed (Reversión completada)

ID de ejecución

Un identificador único que Generador de Imágenes asigna para realizar un seguimiento de los recursos de tiempo de ejecución cada vez que se ejecuta un flujo de trabajo.

Inicio

La marca de tiempo cuando se inició la instancia de tiempo de ejecución de este flujo de trabajo.

Finalización

La marca de tiempo cuando finalizó la instancia de tiempo de ejecución de este flujo de trabajo.

Total de pasos

Número total de pasos en el flujo de trabajo. Debe ser igual a la suma de los recuentos de pasos que se realizaron correctamente, que se omitieron y que fallaron.

Pasos realizados correctamente

Un recuento del tiempo de ejecución del número de pasos del flujo de trabajo que se ejecutaron correctamente.

Pasos con error

Un recuento del tiempo de ejecución del número de pasos del flujo de trabajo que tuvieron errores.

Pasos omitidos

Un recuento del tiempo de ejecución del número de pasos del flujo de trabajo que se omitieron.

Los detalles de la siguiente lista indican el estado actual de todos los pasos de esta instancia de tiempo de ejecución del flujo de trabajo. Generador de Imágenes muestra los mismos detalles para todos los tipos de imágenes.

N.° de paso

Un número que representa el orden en el que Generador de Imágenes ejecuta los pasos del flujo de trabajo.

ID de paso

Un identificador único para el paso del flujo de trabajo, que se asigna en el tiempo de ejecución.

Estado del paso

El estado de tiempo de ejecución actual del paso del flujo de trabajo especificado.

Estado de reversión

El estado de reversión actual si se produjo un error en esta instancia de tiempo de ejecución del flujo de trabajo.

Nombre del paso

El nombre del paso del flujo de trabajo especificado.

Inicio

La marca de tiempo cuando se inició el paso especificado para esta instancia de tiempo de ejecución del flujo de trabajo.

Finalización

La marca de tiempo cuando finalizó el paso especificado para esta instancia de tiempo de ejecución del flujo de trabajo.

Pestaña de resultados de seguridad

Si activó el escaneo, la pestaña Resultados de seguridad muestra los resultados sobre vulnerabilidades y exposiciones comunes (CVE). Amazon Inspector identificó estos resultados en la instancia de prueba que Generador de imágenes lanzó para crear la nueva imagen. Para garantizar que Generador de Imágenes capture los resultados de la imagen, debe configurar el escaneo de la siguiente manera:

1. Active los escaneos de Amazon Inspector para su cuenta. Para obtener más información, consulte Introducción a Amazon Inspector en la Guía del usuario de Amazon Inspector.
2. Active los resultados de seguridad de la canalización que crea esta imagen. Cuando activa los resultados de seguridad para su canalización, Generador de imágenes guarda una instantánea de los resultados antes de finalizar la instancia de prueba. Para obtener más información, consulte Configure los escaneos de seguridad para las imágenes de Image Builder en Consola de administración de AWS

La pestaña Resultados de seguridad incluye los siguientes detalles de cada vulnerabilidad que Amazon Inspector identificó para su imagen:

Gravedad

El nivel de gravedad del resultado de CVE. Los valores son los siguientes:

- Sin clasificar
- Informativo
- Bajo
- Medio
- Alto

- Crítica

ID del resultado

Identificador único del resultado de CVE que Amazon Inspector detectó para su imagen al analizar la instancia de prueba. El ID está vinculado a la página Resultados de seguridad > Por vulnerabilidad. Para obtener más información, consulte Gestione los resultados de seguridad de las imágenes de Image Builder en Consola de administración de AWS.

Origen

La fuente de la información de vulnerabilidad para el resultado de CVE.

Antigüedad

El número de días transcurridos desde que se observó el resultado por primera vez en su imagen.

Puntuación de Inspector

La puntuación que Amazon Inspector asignó al resultado de CVE.

Pestaña Etiquetas

La pestaña Etiquetas muestra todas las etiquetas que haya definido para la imagen.

Obtenga los detalles de la política de imágenes en el AWS CLI

En el siguiente ejemplo, se muestra cómo obtener los detalles de una política de imágenes con su nombre de recurso de Amazon (ARN).

```
aws imagebuilder get-image-policy --image-arn arn:aws:imagebuilder:us-
west-2:123456789012:image/example-image/2019.12.02
```

Creación de imágenes personalizadas con Generador de imágenes

Existen varias formas diferentes de crear una nueva imagen de Generador de Imágenes. Por ejemplo, puede utilizar uno de los métodos siguientes para crear una imagen con Consola de administración de AWS o AWS CLI. También puedes usar la acción de la CreateImageAPI o ejecutar una canalización de compilación para crear la imagen. Para ver la solicitud de SDK asociada a la acción de la API, puede consultar el enlace Vea también de ese comando en la Referencia de la API de EC2 Image Builder.

Consola de administración de AWS

Para crear una imagen a partir de una canalización existente, puede ejecutar la canalización manualmente de la siguiente manera. También puede utilizar el asistente para canalizaciones para crear una nueva imagen desde cero. Consulte Asistente de canalización: crear AMI o Asistente de canalización: crear una imagen de contenedor, según el tipo de imagen que desee crear.

1. Abra la consola EC2 de Image Builder en https://console.aws.amazon.com/imagebuilder/.
2. En el panel de navegación, elija Canalizaciones de imágenes.
3. Seleccione la casilla de verificación junto al nombre de la canalización que desea ejecutar.
4. Para crear la imagen, seleccione Ejecutar canalización en el menú Acciones. Esto inicia la canalización.

También puedes especificar un cronograma para ejecutar tu canalización o usar Amazon EventBridge para ejecutar tu canalización en función de las reglas que configures.

AWS CLI

Antes de ejecutar el create-image comando en el AWS CLI, debes crear los siguientes recursos si aún no existen:

Recursos necesarios de

- Receta: debe especificar exactamente una receta para su imagen, de la siguiente manera:

  Receta de imagen

  Especifique el nombre de recurso de Amazon (ARN) para su recurso de receta de imágenes con el parámetro `--image-recipe-arn`.

  Receta de contenedor

  Especifique el ARN del recurso de recetas de contenedores con el parámetro `--container-recipe-arn`.
- Configuración de infraestructura: especifique el ARN del recurso de configuración de infraestructura con el parámetro `--infrastructure-configuration-arn`.

También puede especificar cualquiera de las siguientes fuentes de recursos que necesite la imagen:

Recursos y configuración opcionales

- Ajustes de distribución: de forma predeterminada, Generador de Imágenes distribuye el recurso de imagen de salida a su cuenta en la región en la que ejecuta el comando create-image. Para proporcionar destinos o configuraciones adicionales para su distribución, especifique el ARN del recurso de configuración de distribución con el parámetro `--distribution-configuration-arn`.
- Análisis de imágenes: para configurar las instantáneas de los resultados de Amazon Inspector en la instancia de prueba de su imagen o contenedor, utilice el parámetro `--image-scanning-configuration`. En el caso de las imágenes de contenedores, también debe especificar el repositorio de ECR que Amazon Inspector utiliza para sus escaneos.
- Pruebas de imagen: para suprimir la etapa de prueba de Generador de Imágenes, utilice el parámetro `--image-tests-configuration`. Como alternativa, puede establecer un tiempo de espera durante el cual se puede ejecutar.
- Etiquetas de imagen: utilice el parámetro `--tags` para añadir etiquetas al recurso de imagen de salida.
- Flujos de trabajo de imágenes: si no especifica ningún flujo de trabajo de creación o prueba, Generador de imágenes crea la imagen con el flujo de trabajo de imágenes predeterminado. Para especificar los flujos de trabajo que ha creado, utilice el parámetro `--workflows`.

> **Note**
>
> Si especifica flujos de trabajo de imágenes, también debe proporcionar el nombre o el ARN del rol de IAM que Generador de imágenes utiliza para ejecutar las acciones de flujo de trabajo en el parámetro `--execution-role`.

El siguiente ejemplo muestra cómo crear una imagen con el comando create-image de AWS CLI . Para obtener más información, consulte Referencia de comandos de la AWS CLI .

Ejemplo: creación de una imagen básica con distribución predeterminada

```
aws imagebuilder create-image --image-recipe-arn arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/simple-recipe-linux/1.0.0 --infrastructure-configuration-arn arn:aws:imagebuilder:us-west-2:123456789012:infrastructure-configuration/simple-infra-config-linux
```

Salida:

```
{
"requestId": "1abcd234-e567-8fa9-0123-4567b890cd12",
"imageVersionList": [
    {
        "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image/simple-recipe-
linux/1.0.0",
        "name": "simple-recipe-linux",
        ...
    }
]
}
```

Cancela la creación de imágenes desde AWS CLI

Para cancelar la creación de una imagen en curso, utilice el comando cancel-image-creation siguiente:

```
aws imagebuilder cancel-image-creation --image-build-version-arn
 arn:aws:imagebuilder:us-west-2:123456789012:image/my-example-recipe/2019.12.03/1
```

Importación y exportación de imágenes de máquinas virtuales con Generador de imágenes

Al exportar una máquina virtual desde su entorno de virtualización, ese proceso crea un conjunto de uno o más archivos contenedores de discos que actúan como instantáneas del entorno, la configuración y los datos de la máquina virtual. Puede usar estos archivos para importar su máquina virtual y utilizarlos como imagen base para sus recetas de imágenes. Para exportar, puede crear archivos de disco de máquina virtual como salida de su compilación de imágenes personalizada y distribuir los archivos.

Generador de Imágenes admite los siguientes formatos de archivo para los contenedores de discos de máquinas virtuales:

- Archivo de virtualización abierto (OVA)
- Virtual Machine Disk (VMDK)
- Virtual Hard Disk (VHD/VHDX)
- Raw

La importación utiliza los discos para crear una imagen de máquina de Amazon (AMI) y un recurso de imagen de Generador de Imágenes, cualquiera de los cuales puede servir como imagen base para su receta de imagen personalizada. Los discos de las máquinas virtuales deben almacenarse en buckets de S3 para la importación. También puede importar desde una instantánea de EBS existente.

En la consola de Generador de Imágenes, puede importar la imagen directamente y, a continuación, utilizar la imagen de salida o la AMI en sus recetas, o bien puede especificar los parámetros de importación al crear la receta o la versión de la receta. Para obtener más información sobre la importación como parte de su receta de imagen, consulte Configuración de la importación de máquinas virtuales.

Importación de una máquina virtual a Generador de imágenes

Image Builder se integra con la Import/Export API de Amazon EC2 VM para permitir que el proceso de importación se ejecute de forma asíncrona en segundo plano. Generador de Imágenes hace referencia al ID de la tarea desde la importación de la máquina virtual para realizar un seguimiento de su progreso y crea un recurso de imagen de Generador de Imágenes como salida. Esto le permite hacer referencia al recurso de imagen de Generador de Imágenes en sus recetas antes de que finalice la importación de la máquina virtual.

Console

Para importar una VM con la consola Generador de Imágenes, siga estos pasos:

1. Abra la consola EC2 de Image Builder en https://console.aws.amazon.com/imagebuilder/.
2. Elija Imágenes en el panel de navegación.
3. Para abrir el cuadro de diálogo de importación, seleccione Importar imagen.
4. Introduzca la siguiente información general:
   - Especifique un nombre único para la imagen.
   - Especifique una versión para la imagen base. Use el siguiente formato: *major.minor.patch*.
5. Elija el tipo de importación: importación de máquina virtual.
6. Proporcione los detalles de cada una de las siguientes secciones de la página de Importación de imágenes. A continuación, elija Importar imagen cuando haya terminado.

Sistema operativo de imagen base

1. Seleccione la opción del sistema operativo de imágenes (SO) que coincida con la plataforma de SO de su máquina virtual.
2. Seleccione en la lista la versión del SO que coincida con la versión de su máquina virtual.

Configuración de la importación de máquinas virtuales

1. Al exporta la máquina virtual desde su entorno de virtualización, ese proceso crea un conjunto de uno o más archivos contenedores de discos. Actúan como instantáneas del entorno, los ajustes y los datos de la máquina virtual. Puede utilizar estos archivos para importar su máquina virtual como imagen base para su receta de imagen. Para obtener más información sobre la importación VMs en Image Builder, consulteImportación y exportación de imágenes de máquinas virtuales.

   Para especificar la ubicación del origen de importación, siga estos pasos:

   Importación de fuente

   Especifique la fuente del primer contenedor de discos o instantáneas de imágenes de máquina virtual que se va a importar en la sección Contenedor de discos 1.

   a. Origen: puede ser un bucket de S3 o una instantánea de EBS.
   b. Seleccione la ubicación S3 del disco: introduzca la ubicación en Amazon S3 en la que se almacenan las imágenes del disco. Para buscar la ubicación, elija Examinar S3.
   c. Para agregar un contenedor de discos, elija Agregar contenedor de disco.

2. rol de IAM

   Para asociar un rol de IAM a su configuración de VM Import, seleccione el rol de la lista desplegable del rol de IAM o elija Crear un nuevo rol para crear uno nuevo. Si crea un rol nuevo, la página de la consola de roles de IAM se abre en una pestaña independiente.

3. Configuración avanzada: opcional

   Los siguientes ajustes son opcionales. Con estos ajustes, puede configurar el cifrado, las licencias, las etiquetas y mucho más para la imagen base que se crea con la importación.

   Arquitectura de la imagen base

   Para especificar la arquitectura del origen de importación de la máquina virtual, seleccione un valor de la lista de arquitecturas.

   Cifrado

   Si las imágenes de disco de la máquina virtual están cifradas, debe proporcionar una clave para utilizarla en el proceso de importación. Para especificar una clave de KMS para la importación, seleccione un valor de la lista de Cifrado (clave de KMS). La lista contiene las claves KMS a las que tiene acceso su cuenta en la región actual.

   Administración de licencias

   Al importar una máquina virtual, el proceso de importación detecta automáticamente el SO de la máquina virtual y aplica la licencia adecuada a la imagen base. Según la plataforma del sistema operativo, los tipos de licencia son los siguientes:

   - Licencia incluida: se aplica a la imagen base una licencia de AWS adecuada para su plataforma.
   - Traiga su propia licencia (BYOL): conserva la licencia de su máquina virtual, si corresponde.

   Para adjuntar las configuraciones de licencia creadas con AWS License Manager la imagen base, seleccione un nombre de la configuración de licencia en la lista. Para obtener más información sobre License Manager, consulte Trabajar con AWS License Manager

   > **Note**
   >
   > - Las configuraciones de la licencia contienen reglas de asignación de licencias que se basan en las condiciones de los contratos de su empresa.
   > - Linux solo admite licencias BYOL.

Etiquetas (imagen base)

Las etiquetas utilizan pares clave-valor para asignar texto con capacidad de búsqueda a su recurso de Generador de Imágenes. Para especificar etiquetas para la imagen base importada, introduzca los pares clave-valor mediante los cuadros Clave y Valor.

Para agregar una etiqueta, elija Add tag (Añadir etiqueta). Para quitar una etiqueta, elija Remove tag (Eliminar etiqueta).

AWS CLI

Para importar una máquina virtual de los discos a una AMI y crear un recurso de imagen de Generador de Imágenes al que pueda hacer referencia de inmediato, siga estos pasos desde la AWS CLI:

1. Inicie una importación de máquinas virtuales con el Import/Export import-image comando Amazon EC2 VM en AWS CLI. Anote el ID de la tarea que se devuelve en la respuesta del comando. Lo necesitará para el siguiente paso. Para obtener más información, consulte Importación de una máquina virtual como imagen mediante VM Import/Export en la Guía del Import/Export usuario de máquinas virtuales.

2. Creación de un archivo JSON de entrada de la CLI

   Para simplificar el import-vm-image comando Image Builder que se utiliza en el AWS CLI, creamos un archivo JSON que contiene toda la configuración de importación que queremos pasar al comando.

   > **Note**
   >
   > La convención de nomenclatura de los valores de datos del archivo JSON sigue el patrón que se especifica para los parámetros de solicitud de operación de la API Image Builder. Para revisar los parámetros de la solicitud de operación de la API, consulte la ImportVmImageoperación en la Referencia de la API de EC2 Image Builder.
   > Para proporcionar los valores de los datos como parámetros de la línea de comandos, consulte los nombres de los parámetros especificados en la referencia

de comandos de la AWS CLI para el comando import-vm-image de Generador de Imágenes como opciones.

Este es un resumen de los parámetros que especificamos en este ejemplo:

- Nombre (cadena, obligatorio): nombre del recurso de imagen de Generador de Imágenes que se creará como resultado de la importación.
- semanticVersion (cadena, obligatorio): la versión semántica de la imagen de salida que especifica la versión en el siguiente formato, con valores numéricos en cada posición para indicar una versión específica: <major>.<minor>.<patch>. Por ejemplo, `1.0.0`. Para obtener más información sobre el control de versiones semántico para los recursos de Generador de Imágenes, consulte Control de versiones semántico en Generador de imágenes.
- Descripción (cadena): descripción de la receta de imagen.
- Plataforma (cadena, obligatoria): la plataforma del sistema operativo de la máquina virtual importada.
- vmImportTaskId (cadena, obligatorio): el `ImportTaskId` (AWS CLI) del proceso de importación de Amazon EC2 VM. Generador de Imágenes monitorea el proceso de importación para incluir la AMI que crea y compilar un recurso de imagen de Generador de Imágenes que se pueda utilizar en recetas de forma inmediata.
- Etiquetas (mapa de cadenas): las etiquetas son pares clave-valor que se adjuntan a los recursos de importación. Se permiten hasta 50 pares clave-valor.

Guarde el archivo como `import-vm-image.json` para usarlo en el comando import-vm-image de Generador de Imágenes.

```
{
    "name": "example-request",
    "semanticVersion": "1.0.0",
    "description": "vm-import-test",
    "platform": "Linux",
    "vmImportTaskId": "import-ami-01ab234567890cd1e",
    "tags": {
     "Usage": "VMIE"
    }
}
```

3. Cómo importar la imagen

   Ejecute el comando import-vm-image, con el archivo que creó como entrada:

   ```
   aws imagebuilder import-vm-image --cli-input-json file://import-vm-image.json
   ```

   > **Note**
   >
   > - Debe incluir la notación `file://` al principio de la ruta del archivo JSON.
   > - La ruta del archivo JSON debe seguir la convención apropiada para el sistema operativo base donde se está ejecutando el comando. Por ejemplo, Windows utiliza la barra diagonal inversa (\) para referirse a la ruta de directorios, mientras que Linux y macOS usan la barra diagonal (/).

Distribuya los discos de las máquinas virtuales de la compilación de la imagen desde AWS CLI

Puede configurar la distribución de los archivos con formato de disco de máquina virtual compatibles en los buckets de S3 de las regiones de destino como parte de su proceso habitual de compilación de imágenes, mediante los ajustes de distribución de Generador de Imágenes en la AWS CLI. Para obtener más información, consulte Ejemplo: cree la configuración de distribución para los discos de máquinas virtuales de salida desde AWS CLI.

Importe imágenes de disco ISO de Windows verificadas con Image Builder

Un archivo ISO del sistema operativo Windows es un archivo de imagen de disco que contiene el paquete de instalación completo para una versión específica del sistema operativo Windows. Microsoft proporciona los archivos ISO oficiales del sistema operativo Windows para su descarga, ya sea directamente desde su sitio web o a través de distribuidores autorizados. Es importante asegurarse de obtener los archivos ISO de una fuente confiable y legítima para evitar posibles programas maliciosos o versiones no autorizadas.

EC2 Image Builder utiliza el `build-image-from-iso` flujo de trabajo de importación para importar el archivo de disco ISO y crear un volumen secundario a partir de él. Una vez completada la

configuración, Image Builder toma una instantánea del volumen que creó a partir de la importación y la utiliza para crear una Amazon Machine Image (AMI).

Sistemas operativos compatibles para la importación de imágenes de disco ISO

Image Builder admite las siguientes imágenes de disco ISO del sistema operativo Windows:

- Windows 11 Enterprise versión 25H2 (x64)
- Windows 11 Enterprise versión 24H2 (x64)
- Windows 11 Enterprise versión 23H2 (x64)

Image Builder no admite las siguientes imágenes de disco ISO del sistema operativo Windows:

- Imágenes del canal de servicio a largo plazo (LTSC)
- Imágenes de disco ISO creadas con la herramienta de creación de Windows Media
- Imágenes de evaluación

Requisitos previos para importar una imagen de disco ISO

Para importar una imagen de disco ISO, primero debe cumplir los siguientes requisitos previos:

- El sistema operativo de la imagen de disco debe ser compatible con Image Builder. Para obtener una lista de los sistemas operativos compatibles, consulteSistemas operativos compatibles para la importación de imágenes de disco ISO.
- Para asegurarte de que puedes importar tu imagen ISO, descárgala del centro de administración de Microsoft 365.
- Antes de poder ejecutar el proceso de importación, debe cargar el archivo de disco ISO en Amazon S3 en el mismo Cuenta de AWS Región de AWS lugar donde se realiza la importación.
- La extensión del archivo distingue mayúsculas de minúsculas para el proceso de importación y debe serlo `.ISO`. Si la extensión del archivo está en minúsculas, puede ejecutar uno de los siguientes comandos para cambiarle el nombre:

Command

```
aws s3 cp s3://amzn-s3-demo-bucket/Win11_24H2_English.iso s3://amzn-s3-demo-
bucket/Win11_24H2_English.ISO
```

PowerShell

```
Copy-S3Object -BucketName amzn-s3-demo-bucket -Key Win11_24H2_English.iso -
DestinationKey Win11_24H2_English.ISO
```

- Las licencias de Microsoft no se incluyen automáticamente en la importación. Debe traer su propia licencia (BYOL). Para obtener más información sobre las licencias de software de Microsoft, consulte Licencias en la página de preguntas frecuentes de Amazon Web Services y Microsoft.
- El proceso de importación utiliza dos funciones de IAM independientes, como se indica a continuación:

  Rol de ejecución

  Esta función otorga permiso para que Image Builder llame Servicios de AWS en su nombre. Puede especificar el rol AWSServiceRoleForImageBuildervinculado al servicio, que incluye los permisos necesarios para el rol de ejecución, o puede crear su propio rol.

  Rol de perfil de instancia

  Este rol otorga permiso para las acciones que el servicio realiza en la EC2 instancia. Puede especificar un rol de perfil de instancia en su recurso de configuración de infraestructura. Adjunta las siguientes políticas administradas a tu rol de perfil de instancia para asegurarte de que tienes todos los permisos necesarios para el proceso de importación.

  - EC2InstanceProfileForImageBuilder
  - AmazonSSMManagedInstanceCore

  Para obtener más información, consulte Administre la configuración de la infraestructura de Generador de imágenes.

Importación de una imagen de disco ISO a Image Builder

Antes de iniciar el proceso de importación, asegúrese de cumplir con todos losRequisitos previos.

El proceso de importación instala el software y los controladores siguientes en la imagen:

- EC2Inicie la versión 2
- AWS Systems Manager agente
- AWS NVMe conductor
- AWS controlador de red ENA
- AWS Controlador serie PCI
- EC2 Controlador de utilidad de Windows
- Kit de actualización de Microsoft Defender

El proceso de importación realiza las siguientes actualizaciones de configuración en la imagen:

- Configura el sistema para que utilice el servidor Amazon Time.

Console

Para importar una imagen de disco ISO con la consola Image Builder, siga estos pasos:

1. Abra la consola EC2 de Image Builder en https://console.aws.amazon.com/imagebuilder/.
2. Elija Imágenes en el panel de navegación.
3. Para abrir el cuadro de diálogo de importación, seleccione Importar imagen.
4. Introduzca la siguiente información general:
   - Especifique un nombre único para la imagen.
   - Especifique una versión para la imagen base. Use el siguiente formato: *major*.*minor*.*patch*.
5. Elija el tipo de importación: importación ISO.
6. Introduzca los siguientes detalles de configuración de importación ISO. A continuación, elija Importar imagen cuando haya terminado.
   - URI de S3: introduzca la ubicación en la que está almacenado el archivo de disco ISO. Para buscar el archivo, seleccione Browse S3.
   - Función de IAM: para asociar una función de IAM a la configuración de importación, seleccione la función en la lista desplegable de funciones de IAM o seleccione Crear nueva función para crear una nueva. Si crea un rol nuevo, la página de la consola de roles de IAM se abre en una pestaña independiente.

Puede especificar el rol AWSServiceRoleForImageBuildervinculado al servicio o puede especificar su propio rol personalizado para el acceso al servicio.

7. Si lo desea, puede añadir etiquetas a su recurso de imágenes de Image Builder. Esto no añade las etiquetas a la AMI.
8. La configuración de la infraestructura ISO define los ajustes de la instancia que Image Builder lanza para alojar el proceso de importación. Puede usar una configuración de infraestructura que cree Image Builder, en función de los valores predeterminados del servicio, o puede usar una configuración de infraestructura existente. Para obtener más información, consulte Administre la configuración de la infraestructura de Generador de imágenes.

   Para crear una nueva configuración de infraestructura, elija Crear configuración de infraestructura. Esto se abre en una pestaña independiente. Cuando haya terminado de crear el nuevo recurso, puede volver a la configuración de importación y elegir Usar la configuración de infraestructura existente.
9. Para iniciar el proceso de importación, selecciona Importar imagen.

Una vez finalizada la importación, la imagen aparecerá en la lista de imágenes de tu propiedad. Para obtener más información, consulta Enumeración de imágenes.

AWS CLI

En este ejemplo se muestra cómo importar una imagen de un archivo de disco ISO y crear una AMI a partir de ella con AWS CLI.

Este es un resumen de los parámetros que especificamos en este ejemplo:

- Nombre (cadena, obligatorio): nombre del recurso de imagen de Generador de Imágenes que se creará como resultado de la importación.
- semanticVersion (cadena, obligatorio): la versión semántica de la imagen de salida que especifica la versión en el siguiente formato, con valores numéricos en cada posición para indicar una versión específica: <major>.<minor>.<patch>. Por ejemplo, `1.0.0`. Para obtener más información sobre el control de versiones semántico para los recursos de Generador de Imágenes, consulte Control de versiones semántico en Generador de imágenes.
- Descripción (cadena): descripción de la receta de imagen.
- executionRole (cadena): el nombre o el nombre del recurso de Amazon (ARN) del rol de IAM que concede a Image Builder acceso para realizar acciones de flujo de trabajo para importar una imagen de un archivo ISO de Microsoft. Puede especificar el rol

AWSServiceRoleForImageBuildervinculado al servicio o puede especificar su propio rol personalizado para el acceso al servicio.

- platform (cadena, obligatoria): la plataforma del sistema operativo para la imagen de disco ISO. Los valores válidos incluyen `Windows`.
- osVersion (cadena, obligatoria): la versión del sistema operativo de la imagen de disco ISO. Los valores válidos incluyen `Microsoft Windows 11`.
- infrastructureConfigurationArn(string, required): el nombre del recurso de Amazon (ARN) del recurso de configuración de infraestructura que se utiliza para lanzar la EC2 instancia en la que se crea la imagen ISO.
- uri (cadena, obligatorio): el URI del archivo de disco ISO que está almacenado en Amazon S3.

```
aws imagebuilder import-disk-image \
    --name "example-iso-disk-import" \
    --semantic-version "1.0.0" \
    --description "Import an ISO disk image" \
    --execution-role "AWSServiceRoleForImageBuilder" \
    --platform "Windows" \
    --os-version "Microsoft Windows 11" \
    --infrastructure-configuration-arn "arn:aws:imagebuilder:us-
east-1:111122223333:infrastructure-configuration/example-infrastructure-
configuration-123456789abc",
    --uri: "s3://amzn-s3-demo-source-bucket/examplefile.iso"
```

Una vez finalizada la importación, la imagen aparecerá en la lista de imágenes de su propiedad. Para obtener más información, consulta Enumeración de imágenes.

PowerShell

En este ejemplo se muestra cómo importar una imagen de un archivo de disco ISO y crear una AMI a partir de ella PowerShell.

Este es un resumen de los parámetros que especificamos en este ejemplo:

- Nombre (cadena, obligatorio): nombre del recurso de imagen de Generador de Imágenes que se creará como resultado de la importación.
- semanticVersion (cadena, obligatorio): la versión semántica de la imagen de salida que especifica la versión en el siguiente formato, con valores numéricos en cada posición para indicar una versión específica: <major>.<minor>.<patch>. Por ejemplo, `1.0.0`. Para obtener

más información sobre el control de versiones semántico para los recursos de Generador de Imágenes, consulte Control de versiones semántico en Generador de imágenes.

- Descripción (cadena): descripción de la receta de imagen.
- executionRole (cadena): el nombre o el nombre del recurso de Amazon (ARN) del rol de IAM que concede a Image Builder acceso para realizar acciones de flujo de trabajo para importar una imagen de un archivo ISO de Microsoft. Puede especificar el rol AWSServiceRoleForImageBuildervinculado al servicio o puede especificar su propio rol personalizado para el acceso al servicio.
- platform (cadena, obligatoria): la plataforma del sistema operativo para la imagen de disco ISO. Los valores válidos incluyen `Windows`.
- osVersion (cadena, obligatoria): la versión del sistema operativo de la imagen de disco ISO. Los valores válidos incluyen `Microsoft Windows 11`.
- infrastructureConfigurationArn(string, required): el nombre del recurso de Amazon (ARN) del recurso de configuración de infraestructura que se utiliza para lanzar la EC2 instancia en la que se crea la imagen ISO.
- uri (cadena, obligatorio): el URI del archivo de disco ISO que está almacenado en Amazon S3.

```
Import-EC2IBDiskImage `
    -Name "example-iso-disk-import" `
    -SemanticVersion "1.0.0" `
    -Description "Import an ISO disk image" `
    -ExecutionRole "AWSServiceRoleForImageBuilder" `
    -Platform "Windows" `
    -OsVersion "Microsoft Windows 11" `
    -InfrastructureConfigurationArn "arn:aws:imagebuilder:us-
east-1:111122223333:infrastructure-configuration/example-infrastructure-
configuration-123456789abc" `
    -Uri "s3://amzn-s3-demo-source-bucket/examplefile.ISO"
```

Una vez finalizada la importación, la imagen aparecerá en la lista de imágenes de su propiedad. Para obtener más información, consulta Enumeración de imágenes.

Lanza una instancia desde la AMI de salida

Al lanzar una instancia desde una AMI creada por el proceso de importación, el sistema operativo Windows ejecuta Sysprep Specialize, que descarga e instala automáticamente EC2 Launch v2 y el

Agente de Systems Manager desde puntos de conexión S3 públicos. Estos puntos finales requieren acceso público a Internet. Si lanza una instancia desde una subred privada, el proceso Sysprep Specialize no podrá acceder a los puntos finales de S3 y se producirá un error en el lanzamiento.

Administración de los resultados de seguridad para las imágenes de Generador de Imágenes

Cuando activa el escaneo de seguridad con Amazon Inspector, escanea continuamente las imágenes de las máquinas y las instancias en ejecución en su cuenta para detectar vulnerabilidades en el sistema operativo y el lenguaje de programación. Si está activado, el análisis de seguridad es automático y Generador de imágenes puede guardar una instantánea de los resultados de las instancias de prueba al crear una imagen nueva. Amazon Inspector es un servicio de pago.

Cuando Amazon Inspector descubre vulnerabilidades en el software o en la configuración de la red, toma las siguientes medidas:

- Le notifica que se ha producido un resultado.
- Califica la gravedad del resultado. La clasificación de gravedad categoriza las vulnerabilidades para ayudarle a priorizar sus resultados e incluye los siguientes valores:
  - Sin clasificar
  - Informativo
  - Bajo
  - Medio
  - Alto
  - Crítica
- Proporciona información sobre el resultado y enlaces a recursos adicionales para obtener más detalles.
- Ofrece una guía de corrección para ayudarle a resolver los problemas que generaron el resultado.

Configure los escaneos de seguridad para las imágenes de Image Builder en Consola de administración de AWS

Si has activado Amazon Inspector para tu cuenta, Amazon Inspector escanea automáticamente las EC2 instancias que lanza Image Builder para crear y probar una nueva imagen. Esas instancias

tienen una vida útil corta durante el proceso de compilación y prueba y, por lo general, sus resultados caducan en cuanto se cierran esas instancias. Para ayudarlo a investigar y corregir los resultados de la nueva imagen, tiene la opción de que Generador de imágenes guarde como instantánea los resultados que Amazon Inspector haya identificado para la instancia de prueba durante el proceso de creación.

Paso 1: activar los escaneos de seguridad de Amazon Inspector para su cuenta

Para activar los escaneos de seguridad de Amazon Inspector para su cuenta desde la consola Generador de Imágenes, siga estos pasos:

1. Abra la consola EC2 de Image Builder en https://console.aws.amazon.com/imagebuilder/.
2. En el panel de navegación, elija Ajustes del escaneo de seguridad. Esto abre el cuadro de diálogo de escaneo de seguridad.

   El cuadro de diálogo muestra el estado del escaneo para su cuenta. Si Amazon Inspector ya está activado en su cuenta, el estado es Activado.
3. Siga los pasos 1 y 2 de las instrucciones para activar el escaneo de Amazon Inspector.

> **Note**
>
> Amazon Inspector incurre en cargos. Para obtener más información, consulte Precios de Amazon Inspector.

Si ha activado el escaneo de su canalización, Generador de Imágenes toma una instantánea de los resultados de su instancia de compilación cuando cree una nueva imagen. De esta forma, puede acceder a los resultados una vez que Generador de Imágenes finalice la instancia de creación.

Paso 2: configurar la canalización para guardar instantáneas de las vulnerabilidades detectadas

Para configurar instantáneas de los resultados de vulnerabilidades para su canalización, siga estos pasos:

1. Abra la consola EC2 de Image Builder en https://console.aws.amazon.com/imagebuilder/.
2. En el panel de navegación, elija Canalizaciones de imágenes.
3. Escoja uno de los siguientes métodos para especificar los detalles de la canalización:

Creación de una nueva canalización

1. En la página Canalizaciones de imágenes, seleccione Crear canalizaciones de imágenes. Esto abre la página Especificar los detalles de la canalización en el asistente de canalización.

Actualización de una canalización existente

1. En la página Canalización de imágenes, seleccione el enlace con el nombre de la canalización que desea actualizar. Esto abre una página detallada de la canalización.

> **Note**
>
> También puede seleccionar la casilla de verificación junto al nombre de la canalización que desee actualizar y, a continuación, elegir Ver detalles.

2. En la página de detalles de la canalización, seleccione Editar canalización en el menú Acción. Esto le lleva a la página Editar canalización.

4. En la sección General del asistente de canalización o en la página Editar canalización, seleccione la casilla Habilitar el escaneo de seguridad.

> **Note**
>
> Si quiere desactivar las instantáneas más adelante, puede editar la canalización para desmarcar la casilla. Esto no desactiva el escaneo de Amazon Inspector en su cuenta. Para desactivar el escaneo de Amazon Inspector, consulte Desactivación de Amazon Inspector en la Guía del usuario de Amazon Inspector.

Gestione los resultados de seguridad de las imágenes de Image Builder en Consola de administración de AWS

Las páginas de la lista de resultados de seguridad muestran información de alto nivel sobre los resultados de sus recursos, con vistas basadas en varios filtros diferentes que pueda aplicar. Cada vista incluye las siguientes opciones en la parte superior para cambiar la vista:

- Todos los resultados de seguridad: esta es la vista predeterminada si seleccione la página Resultados de seguridad en el panel de navegación de la consola de Generador de Imágenes.

- Por vulnerabilidad: esta vista muestra una lista de todos los recursos de imágenes de su cuenta que contienen resultados. El identificador de resultados está vinculado a información más detallada sobre el resultado. Esta información aparece en un panel que se abre en el lado derecho de la página. El panel contiene la información siguiente:
  - Una descripción detallada del resultado.
  - Una pestaña de detalles de resultados. Esta pestaña incluye una descripción general de los resultados, los paquetes afectados, un resumen de consejos de corrección, detalles sobre las vulnerabilidades y las vulnerabilidades relacionadas. El identificador de vulnerabilidades enlaza con información detallada sobre vulnerabilidades en la base de datos nacional de vulnerabilidades.
  - Una pestaña de desglose de puntuaciones. Esta pestaña incluye una side-by-side comparación de las puntuaciones de CVSS y Amazon Inspector para que pueda ver dónde Amazon Inspector ha modificado una puntuación, si procede.
- Por canalización de imágenes: esta vista muestra el número de resultados de cada canalización de imágenes de su cuenta. Generador de Imágenes muestra el resumen de los resultados de gravedad media y alta, además de un total de todos los resultados. Todos los datos de la lista están enlazados de la siguiente manera:
  - La columna del nombre de la canalización de imágenes enlaza con la página de detalles de la canalización de imágenes especificada.
  - Los enlaces de la columna de nivel de gravedad abren la vista Todos los resultados de seguridad, filtrada por el nombre y el nivel de gravedad de la canalización de imágenes asociados.

  También puede usar criterios de búsqueda para mejorar los resultados.
- Por imagen: esta vista muestra el número de resultados de cada imagen creada en su cuenta. Generador de Imágenes muestra el resumen de los resultados de gravedad media y alta, además de un total de todos los resultados. Todos los datos de la lista están enlazados de la siguiente manera:
  - La columna nombre de la imagen enlaza con la página de detalles de la imagen de la creación de imagen especificada. Para obtener más información, consulte Ver detalles de recurso de imagen.
  - Los enlaces de la columna de nivel de gravedad abren la vista Todos los resultados de seguridad, filtrados por el nombre de la creación de la imagen y el nivel de gravedad asociados.

  También puede usar criterios de búsqueda para mejorar los resultados.

Generador de Imágenes muestra los siguientes detalles en la sección Lista de resultados de la vista predeterminada Todos los resultados de seguridad.

Gravedad

El nivel de gravedad del resultado de CVE. Los valores son los siguientes:

- Sin clasificar
- Informativo
- Bajo
- Medio
- Alto
- Crítica

ID del resultado

El identificador único del resultado de CVE que Amazon Inspector detectó para su imagen al escanear la instancia de compilación. El ID está vinculado a la página Resultados de seguridad > Por vulnerabilidad.

ARN de imagen

El nombre de recurso de Amazon (ARN) de la imagen con el resultado especificado en la columna Identificador del resultado.

Canalización

La canalización que creó la imagen especificada en la columna ARN de la imagen.

Descripción

Una breve descripción del resultado.

Puntuación de Inspector

La puntuación que Amazon Inspector asignó al resultado de CVE.

Corrección

Enlaces a detalles sobre el curso de acción recomendado para corregir el resultado.

Fecha de publicación

La fecha y la hora en las que esta vulnerabilidad se agregó por primera vez a la base de datos del proveedor.

Limpieza de los recursos de Generador de imágenes

Para evitar cargos inesperados, asegúrese de limpiar los recursos y canalizaciones que haya creado a partir de los ejemplos de esta guía. Para obtener más información sobre cómo eliminar recursos en Generador de Imágenes, consulte Eliminación de recursos de Generador de imágenes obsoletos o no utilizados.

Administración de políticas de ciclo de vida para imágenes de Generador de imágenes

Al crear imágenes personalizadas, es importante tener un plan para retirarlas antes de que queden obsoletas. Las canalizaciones de Generador de imágenes pueden aplicar actualizaciones y revisiones de seguridad automáticamente. Sin embargo, cada creación crea una nueva versión de la imagen y todos los recursos asociados que distribuye. Las versiones anteriores permanecen en la cuenta hasta que las elimine manualmente o cree un script que se encargue de la tarea.

Con las políticas de administración del ciclo de vida de Generador de imágenes, puede automatizar el proceso de obsolescencia, deshabilitación y eliminación de imágenes obsoletas y sus recursos asociados. Los recursos asociados pueden incluir imágenes de salida que hayas distribuido a otras Cuentas de AWS organizaciones y unidades organizativas (OUs) en todas partes Regiones de AWS. Debe definir las reglas de cómo y cuándo dar cada paso del proceso de ciclo de vida y qué pasos incluir en la política.

Ventajas de la administración automatizada del ciclo de vida

Algunas de las ventajas generales de la administración automatizada del ciclo de vida son las siguientes:

- Simplifica la administración del ciclo de vida de las imágenes personalizadas con una forma automatizada de retirar las imágenes y los recursos asociados.
- Ayuda a prevenir los riesgos de conformidad que se originan en el uso de imágenes desactualizadas para lanzar nuevas instancias.
- Mantiene actualizados los inventarios de imágenes al eliminar las imágenes desactualizadas.
- Puede reducir los costos de almacenamiento y transferencia de datos al eliminar, de forma opcional, los recursos asociados a las imágenes que se eliminan.

Obtención de ahorros de costos

El uso de EC2 Image Builder para crear AMI personalizadas o imágenes de contenedor no tiene costo alguno. Sin embargo, se aplica un precio estándar para otros servicios que se utilizan en el proceso. Al eliminar las imágenes no utilizadas o desactualizadas y los recursos asociados a ellas Cuenta de AWS, puede ahorrar tiempo y costes de las siguientes maneras:

- Reduzca el tiempo que se tarda en aplicar revisiones a las imágenes existentes cuando no esté aplicando revisiones también a las imágenes sin utilizar o desactualizadas.
- En el caso de los recursos de imágenes de AMI que elimine, puede optar por eliminar también las instantáneas distribuidas AMIs y sus instantáneas asociadas. Este enfoque puede ahorrar en el costo de almacenar las instantáneas.
- En el caso de los recursos de imágenes de contenedor que elimine, puede optar por eliminar los recursos subyacentes. Este enfoque puede ahorrar costos de almacenamiento de Amazon ECR y las tarifas de transferencia de datos de las imágenes de Docker almacenadas en los repositorios de ECR.

> **Note**
>
> Generador de imágenes no puede evaluar el impacto potencial de todas las posibles dependencias descendentes, como los grupos de escalado automático o las plantillas de lanzamiento. Debe tener en cuenta las dependencias descendentes de sus imágenes al configurar las acciones de las políticas.

Contenido

- Requisitos previos de administración del ciclo de vida para imágenes de Generador de imágenes
- Enumeración de las políticas de administración del ciclo de vida para recursos de imágenes de Generador de imágenes
- Visualización de detalles de políticas de ciclo de vida
- Creación de políticas de ciclo de vida
- Funcionamiento de las reglas de administración del ciclo de vida para recursos de imagen de Generador de imágenes

Requisitos previos de administración del ciclo de vida para imágenes de Generador de imágenes

Antes de poder definir las políticas y reglas de administración del ciclo de vida de EC2 Image Builder para sus recursos de imágenes, debe cumplir los siguientes requisitos previos.

- Cree un rol de IAM que conceda permiso a Generador de imágenes para ejecutar políticas de ciclo de vida. Para crear la función, consulte Creación de un rol de IAM para la administración del ciclo de vida de Generador de imágenes.
- Cree un rol de IAM en la cuenta de destino para los recursos asociados que se distribuyeron entre las cuentas. El rol otorga permiso para que Generador de imágenes realice acciones del ciclo de vida con los recursos asociados en la cuenta de destino. Para crear la función, consulte Creación de un rol de IAM para la administración del ciclo de vida de Generador de imágenes entre cuentas.

  > **Note**
  >
  > Este requisito previo no se aplica si ha otorgado permisos de lanzamiento para una AMI de salida. Con los permisos de lanzamiento, la cuenta con la que ha compartido es propietaria de las instancias que se lanzan desde la AMI compartida, pero todos los recursos de la AMI permanecen en su cuenta.

- En el caso de las imágenes de contenedor, debe agregar la siguiente etiqueta a los repositorios de ECR para permitir que Generador de imágenes ejecute acciones del ciclo de vida en las imágenes de contenedor almacenadas en el repositorio: `LifecycleExecutionAccess: EC2 Image Builder`.

Creación de un rol de IAM para la administración del ciclo de vida de Generador de imágenes

Para conceder permiso para que Generador de imágenes ejecute políticas de ciclo de vida, primero debe crear el rol de IAM que utiliza para realizar acciones del ciclo de vida. Siga estos pasos para crear el rol de servicio que otorga el permiso.

1. Abra la consola de IAM en https://console.aws.amazon.com/iam/.
2. Elija la opción Roles en el panel de navegación.
3. Elija Crear rol. Así se abre el primer paso del proceso Seleccionar entidad de confianza para crear el rol.
4. Seleccione la opción Política de confianza personalizada en Tipo de entidad de confianza.
5. Copie la siguiente política de confianza JSON y péguela en el área de texto Política de confianza personalizada para sustituir el texto de ejemplo. Esta política de confianza permite a Generador de imágenes asumir el rol que se creó para ejecutar las acciones del ciclo de vida.

JSON

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

6. Seleccione la siguiente política administrada de la lista: EC2ImageBuilderLifecycleExecutionPolicy y, a continuación, elija Siguiente. Se abre la página Nombrar, revisar y crear.

> **Tip**
>
> Filtre por `image` para optimizar los resultados.

7. Escriba un Role name.
8. Después de revisar la configuración, elija Crear rol.

Creación de un rol de IAM para la administración del ciclo de vida de Generador de imágenes entre cuentas

Para conceder permiso para que Generador de imágenes realice acciones del ciclo de vida en las cuentas de destino de los recursos asociados, primero debe crear el rol de IAM que utiliza para realizar acciones del ciclo de vida en esas cuentas. Debe crear el rol en la cuenta de destino.

Siga estos pasos para crear el rol de servicio que otorga el permiso en la cuenta de destino.

1. Abra la consola de IAM en https://console.aws.amazon.com/iam/.
2. Elija la opción Roles en el panel de navegación.
3. Elija Crear rol. Así se abre el primer paso del proceso Seleccionar entidad de confianza para crear el rol.
4. Seleccione la opción Política de confianza personalizada en Tipo de entidad de confianza.
5. Copie la siguiente política de confianza JSON y péguela en el área de texto Política de confianza personalizada para sustituir el texto de ejemplo. Esta política de confianza permite a Generador de imágenes asumir el rol que se creó para ejecutar las acciones del ciclo de vida.

> **Note**
>
> Cuando Generador de imágenes utiliza este rol en la cuenta de destino para actuar sobre los recursos asociados que se distribuyeron entre las cuentas, actúa en nombre del propietario de la cuenta de destino. La Cuenta de AWS que configure como parte de la política de confianza es la cuenta `aws:SourceAccount` en la que Image Builder distribuyó esos recursos.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": [
                    "imagebuilder.amazonaws.com"
                ]
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "444455556666"
                },
                "StringLike": {
                    "aws:SourceArn": "arn:*:imagebuilder:*:*:image/*/*/*"
                }
            }
        }
    ]
```

```
}
```

6. Seleccione la siguiente política administrada de la lista: EC2ImageBuilderLifecycleExecutionPolicy y, a continuación, elija Siguiente. Se abre la página Nombrar, revisar y crear.

> **Tip**
>
> Filtre por `image` para optimizar los resultados.

7. Ingrese `Ec2ImageBuilderCrossAccountLifecycleAccess` como Nombre del rol.

> **Important**
>
> `Ec2ImageBuilderCrossAccountLifecycleAccess` debe ser el nombre de este rol.

8. Después de revisar la configuración, elija Crear rol.

Enumeración de las políticas de administración del ciclo de vida para recursos de imágenes de Generador de imágenes

Puede obtener una lista de sus políticas de administración del ciclo de vida de las imágenes que incluya columnas de detalles clave en la página de lista de políticas del ciclo de vida de la API Image Builder Consola de administración de AWS, o con comandos o acciones en la API Image Builder SDKs, o AWS CLI.

Puede usar uno de los siguientes métodos para enumerar los recursos de la política del ciclo de vida de imágenes de Generador de imágenes en su Cuenta de AWS. Para ver la acción de la API, consulte ListLifecyclePolicieslа Referencia de la API de EC2 Image Builder. Para la solicitud de SDK asociada, consulte el enlace Véase también en la misma página.

Consola de administración de AWS

En la consola se muestran los siguientes detalles de las políticas existentes. Puede seleccionar cualquier columna para cambiar el orden de clasificación de los resultados. La lista de políticas se ordena inicialmente por Nombre de la política. El nombre de la columna del orden de clasificación actual aparece en negrita.

Si tiene más de una página de resultados, se activan las flechas de paginación de la esquina superior derecha del panel. Puede filtrar los resultados por nombre de la política, estado de la política, tipo de imagen de salida y ARN del recurso de imagen con la barra de búsqueda.

- Nombre de la política: nombre de la política.
- Estado de la política: si la política está activa o inactiva.
- Tipo: tipo de la imagen de salida que Generador de imágenes distribuye al crear una nueva versión de la imagen (una imagen de AMI o de contenedor).
- Fecha de la última ejecución: última vez que se ejecutó la política de ciclo de vida.
- Fecha de creación: marca de tiempo de la creación de la política de ciclo de vida.
- ARN: nombre de recurso de Amazon (ARN) del recurso de la política de ciclo de vida.

Para enumerar las políticas del ciclo de vida en Consola de administración de AWS, siga estos pasos:

1. Abra la consola EC2 de Image Builder en https://console.aws.amazon.com/imagebuilder/.
2. En el panel de navegación, seleccione Políticas de ciclo de vida. Aquí se muestra una lista de las políticas del ciclo de vida de las imágenes de cada cuenta.

Acciones disponibles

También puede realizar las siguientes acciones para la política de ciclo de vida desde la página de lista Políticas de ciclo de vida.

Para crear una nueva política de ciclo de vida de imágenes, seleccione Crear política de ciclo de vida. Para obtener información sobre cómo crear una política, consulte Creación de políticas de ciclo de vida.

Para todas las acciones siguientes, primero debe seleccionar la política. Para seleccionar una política, puede activar la casilla situada junto al Nombre de la política.

- Para activar o desactivar la política, seleccione Deshabilitar política o Habilitar política en el menú Acciones.
- Para cambiar la política, seleccione Editar política en el menú Acciones.
- Para eliminar una política, seleccione Eliminar política en el menú Acciones.

- Para crear una nueva política que utilice la política seleccionada como configuración de referencia, seleccione Clonar política en el menú Acciones.

AWS CLI

El siguiente ejemplo de comando muestra cómo utilizar la lista de políticas del AWS CLI ciclo de vida de las imágenes para una imagen específica Región de AWS. Para obtener más información sobre los parámetros y las opciones que puede utilizar con este comando, consulte el list-lifecycle-policiescomando en la Referencia de AWS CLI comandos.

Ejemplo:

```
aws imagebuilder list-lifecycle-policies \
--region us-west-1
```

Salida:

```
{
    "lifecyclePolicySummaryList": [
        {
            "arn": "arn:aws:imagebuilder:us-west-2:111122223333:lifecycle-policy/
sample-lifecycle-policy1",
            "name": "sample-lifecycle-policy1",
            "status": "DISABLED",
            "executionRole": "arn:aws:iam::111122223333:role/sample-lifecycle-role",
            "resourceType": "AMI_IMAGE",
            "dateCreated": "2023-11-07T14:57:01.603000-08:00",
            "tags": {}
        },
        {
            "arn": "arn:aws:imagebuilder:us-west-2:111122223333:lifecycle-policy/
sample-lifecycle-policy2",
            "name": "sample-lifecycle-policy2",
            "status": "ENABLED",
            "executionRole": "arn:aws:iam::111122223333:role/sample-lifecycle-role",
            "resourceType": "AMI_IMAGE",
            "dateCreated": "2023-09-06T10:43:21.436000-07:00",
            "dateLastRun": "2023-11-13T04:43:46.106000-08:00",
            "tags": {}
        },
        {
```

```
            "arn": "arn:aws:imagebuilder:us-west-2:111122223333:lifecycle-policy/
sample-lifecycle-policy3",
            "name": "sample-lifecycle-policy3",
            "status": "ENABLED",
            "executionRole": "arn:aws:iam::111122223333:role/sample-lifecycle-role",
            "resourceType": "AMI_IMAGE",
            "dateCreated": "2023-10-19T15:16:40.046000-07:00",
            "dateUpdated": "2023-10-21T20:07:15.958000-07:00",
            "dateLastRun": "2023-11-12T09:27:45.830000-08:00"
}]}
```

> **Note**
>
> Para usar la opción predeterminada Región de AWS, ejecute este comando sin el `--region` parámetro.

Visualización de detalles de políticas de ciclo de vida

En la página de detalles de la política de ciclo de vida de la consola de Generador de imágenes se incluye una sección de resumen con información adicional agrupada en pestañas. El encabezado de la página es el nombre de la política.

En la página de detalles de la política de ciclo de vida de la consola de Generador de imágenes, puede consultar los detalles de una política de ciclo de vida en particular. También puede usar comandos o acciones con la API Image Builder o AWS CLI para obtener detalles de la política. SDKs

Contenido

- Visualización de los detalles de la política de ciclo de vida en la consola de Generador de imágenes

Visualización de los detalles de la política de ciclo de vida en la consola de Generador de imágenes

La página de detalles de la imagen de la consola de Generador de Imágenes incluye una sección de resumen con información adicional agrupada en pestañas. El encabezado de la página es el nombre y la versión de compilación de la receta que creó la imagen.

Secciones y pestañas de detalles de la consola

- Sección de resumen
- Pestaña Reglas
- Pestaña Ámbito
- RunLog pestaña

Sección de resumen

La sección de resumen abarca el ancho de la página e incluye los siguientes detalles. Estos detalles siempre se muestran.

Estado de la política

Si la política está activa o inactiva.

Tipo

Tipo de la imagen de salida que Generador de imágenes distribuye al crear una nueva versión de la imagen (una imagen de AMI o de contenedor).

Date created (Fecha de creación)

Marca de tiempo de la creación de la política de ciclo de vida.

Fecha de modificación

Última vez que se actualizó la política de ciclo de vida.

Fecha de la última ejecución

Última vez que se ejecutó la política de ciclo de vida.

Rol de IAM

Rol de IAM que utiliza Generador de imágenes para realizar acciones del ciclo de vida.

ARN

Nombre de recurso de Amazon (ARN) del recurso de la política de ciclo de vida.

Descripción

Descripción de la política de ciclo de vida, si se indicó.

Pestaña Reglas

En la pestaña Reglas se muestran las reglas del ciclo de vida que configuró para la política que está consultando. En la pestaña se incluyen los datos siguientes:

- Nombre: nombre de la regla. Estos nombres son estáticos y se basan en las acciones de política que puede configurar.
  - `Deprecation rule`
  - `Disable rule`
  - `Deletion rule`
- Regla: breve descripción de la acción que está configurada para la regla.
- Condiciones de la regla: muestra la configuración de la gestión de recursos asociada, las excepciones a la regla y la configuración de retención, si corresponde.

Para obtener más información acerca de la configuración de reglas, consulte Funcionamiento de las reglas de ciclo de vida.

Pestaña Ámbito

En la pestaña Ámbito se muestran los criterios de selección de recursos que están configurados para la política que está consultando. En la pestaña se incluyen los datos siguientes:

- Filtro: ***`type of filter`*** — El tipo de filtro que utilizó para definir el alcance. El tipo de filtro puede ser uno de los siguientes:
  - `recipes`: recetas que se utilizaron para crear las imágenes a las que se aplica la política de ciclo de vida.
  - `tags`: conjunto de etiquetas que Generador de imágenes utiliza para seleccionar los recursos de imagen a los que se aplica la política de ciclo de vida.
- Una barra de búsqueda: puede filtrar la lista por Nombre para agilizar los resultados que se muestran en la pestaña.
- Nombre: cada fila contiene un nombre o una etiqueta que se ha configurado para los criterios de filtrado.
- Versión: si ha configurado un filtro de recetas, Generador de imágenes muestra la versión de la receta.

RunLog pestaña

Cada vez que ejecuta la política para los recursos configurados, Generador de imágenes guarda los detalles de tiempo de ejecución. Cada fila de la tabla representa una única instancia de tiempo de ejecución. En la pestaña se incluyen los datos siguientes:

- ID de ejecución: identifica la instancia de tiempo de ejecución de la política de ciclo de vida.
- Estado de ejecución: estado de tiempo de ejecución que indica si la acción de la política se está ejecutando actualmente, se ha ejecutado correctamente, ha producido un error o se ha cancelado.
- Recurso afectado: indica si la instancia de tiempo de ejecución identificó algún recurso de imagen para las acciones del ciclo de vida.
- Fecha de inicio: marca de tiempo del inicio de la instancia de tiempo de ejecución.
- Fecha de finalización: marca de tiempo de la finalización de la instancia de tiempo de ejecución.

Creación de políticas de ciclo de vida

Al crear una nueva política de ciclo de vida de EC2 Image Builder, la configuración depende del tipo de imagen para la que esté destinada la política. La acción de la API para crear una política de ciclo de vida para los recursos de imágenes de AMI y los recursos de imágenes de contenedores es la misma (CreateLifecyclePolicy). Sin embargo, la configuración de los recursos de imagen y los recursos asociados es diferente. En esta sección se muestra cómo crear políticas de administración del ciclo de vida para ambos.

> **Note**
>
> Antes de crear una política de ciclo de vida, asegúrese de haber cumplido con todos los Requisitos previos.

Creación de políticas de administración del ciclo de vida para recursos de imagen de AMI de Generador de imágenes

Puede usar uno de los siguientes métodos para crear una política de ciclo de vida de imágenes AMI con Consola de administración de AWS o AWS CLI. También puede utilizar la acción de la CreateLifecyclePolicyAPI. Para la solicitud de SDK asociada, puede consultar el enlace Vea también de ese comando en la Referencia de la API de EC2 Image Builder.

Consola de administración de AWS

Para crear una política de ciclo de vida para los recursos de imágenes de la AMI en Consola de administración de AWS, siga estos pasos:

1. Abra la consola EC2 de Image Builder en https://console.aws.amazon.com/imagebuilder/.
2. En el panel de navegación, elija Políticas de ciclo de vida.
3. Elija Crear política de ciclo de vida.
4. Configure las opciones de la política que se describen en los siguientes procedimientos.
5. Para crear la política de ciclo de vida después de configurar las opciones, elija Crear política.

Defina la configuración General de la política.

1. Seleccione la opción AMI en Tipo de política.
2. Escriba el Nombre de la política.
3. Si lo desea, escriba una Descripción para la política de ciclo de vida.
4. De forma predeterminada, la opción Activar está activada. La configuración predeterminada activa la política de ciclo de vida y la agrega a la programación de inmediato. Para crear una política que esté inicialmente desactivada, puede desactivar la opción Activar.
5. Seleccione el rol de IAM que creó para los permisos de la política de ciclo de vida. Si aún no ha creado este rol, consulte Requisitos previos para obtener más información.

Configure el Alcance de la regla para la política.

En esta sección, se configura la selección de recursos para la política de ciclo de vida, en función del tipo de filtro que utilice.

1. Tipo de filtro: recetas: para aplicar las reglas del ciclo de vida a los recursos de imágenes en función de la receta que los creó, seleccione hasta 50 versiones de recetas para la política.
2. Tipo de filtro: etiquetas: para aplicar las reglas del ciclo de vida a los recursos de imágenes en función de las etiquetas de los recursos, ingrese una lista de hasta 50 pares de clave-valor para que coincida con la política.

Active una o más de las siguientes reglas de ciclo de vida para aplicarlas a los recursos que seleccione la política de ciclo de vida. Si un recurso coincide con más de una regla del ciclo de

vida cuando se ejecuta la política, Generador de imágenes realiza las acciones de las reglas en el siguiente orden: 1) marcar como obsoleto, 2) deshabilitar, 3) eliminar.

Regla de obsolescencia

Establece el estado del recurso de imagen de Generador de imágenes en `Deprecated`. Las canalizaciones de Generador de imágenes siguen ejecutándose para imágenes obsoletas. Si lo desea, puede establecer el tiempo de caducidad de la instancia asociada AMIs sin que ello afecte a su capacidad de lanzar nuevas instancias.

- Recuento de unidades: especifique el valor entero del periodo de tiempo que debe transcurrir después de crear un recurso de imagen para marcarlo como `Deprecated`.
- Unidad: seleccione el intervalo de tiempo que se va a utilizar. El intervalo puede ser `Days`, `Weeks`, `Months` o `Years`.
- Obsoleto AMIs: selecciona la casilla de verificación para marcar el Amazon asociado EC2 AMIs con una fecha de caducidad. AMIs Permanecen disponibles y aún puedes lanzar nuevas instancias a partir de ellas.

Regla de deshabilitación

Establece el estado del recurso de imagen de Generador de imágenes en `Disabled`. Esto impide que se ejecuten canalizaciones de Generador de imágenes para esta imagen. Si lo desea, puede deshabilitar la AMI asociada para evitar el lanzamiento de nuevas instancias.

- Recuento de unidades: especifique el valor entero del periodo de tiempo que debe transcurrir después de crear un recurso de imagen para marcarlo como `Disabled`.
- Unidad: seleccione el intervalo de tiempo que se va a utilizar. El intervalo puede ser `Days`, `Weeks`, `Months` o `Years`.
- Desactivar AMIs: selecciona la casilla de verificación para deshabilitar el Amazon EC2 AMIs asociado. Ya no puede utilizarlas AMIs ni lanzar nuevas instancias desde ellas.

Regla de eliminación

Elimina los recursos de imagen por antigüedad o por recuento. Debe definir el umbral que satisfaga sus necesidades. Cuando un recurso de imagen de Generador de imágenes supera el umbral, se elimina. Si lo desea, puede anular el registro de las instantáneas asociadas AMIs o

eliminar las instantáneas correspondientes. AMIs También puede especificar etiquetas para los recursos que desea retener más allá del umbral.

Al configurar la regla de eliminación por antigüedad, Generador de imágenes elimina el recurso de imagen tras el periodo de tiempo que haya configurado. Por ejemplo, elimine los recursos de imágenes después de 6 meses. Al configurarla por recuento, Generador de imágenes retiene el número más reciente de imágenes que especifique, o lo más cerca posible a ese número, y elimina las versiones anteriores.

- Por antigüedad
  - Recuento de unidades: especifique el valor entero del periodo de tiempo que debe transcurrir después de crear un recurso de imagen para eliminarlo.
  - Unidad: seleccione el intervalo de tiempo que se va a utilizar. El intervalo puede ser `Days`, `Weeks`, `Months` o `Years`.
  - Retener al menos una imagen por receta: active la casilla para mantener el último recurso de imagen disponible para cada versión de la receta a la que afecte esta regla.

  Por recuento
  - Recuento de imágenes: especifique el valor entero del número de recursos de imagen recientes que se deben conservar para cada versión de la receta.
- Anular registro AMIs: selecciona la casilla de verificación para anular el registro de Amazon asociado. EC2 AMIs Ya no puede utilizarlas AMIs ni lanzar nuevas instancias desde ellas.
- Conservar imágenes e instantáneas con las etiquetas asociadas: seleccione la casilla de verificación para introducir una lista de etiquetas para los recursos de imágenes que desee conservar. AMIs Las etiquetas se aplican a los recursos de imágenes y a Amazon EC2 AMIs. Puede ingresar hasta 50 pares de clave-valor.

Etiquetas (opcionales)

Agregue etiquetas a la política de ciclo de vida.

AWS CLI

Para crear una nueva política de ciclo de vida de Generador de imágenes, puede utilizar el comando create-lifecycle-policy de la AWS CLI.

Creación de políticas de administración del ciclo de vida para recursos de imagen de contenedor de Generador de imágenes

Puede utilizar uno de los siguientes métodos para crear una política del ciclo de vida de las imágenes de contenedores con Consola de administración de AWS o AWS CLI. También puedes usar la acción de la CreateLifecyclePolicyAPI. Para la solicitud de SDK asociada, puede consultar el enlace Vea también de ese comando en la Referencia de la API de EC2 Image Builder.

Consola de administración de AWS

Para crear una política de ciclo de vida para los recursos de imágenes de contenedores en el Consola de administración de AWS, sigue estos pasos:

1. Abra la consola EC2 de Image Builder en https://console.aws.amazon.com/imagebuilder/.
2. En el panel de navegación, elija Políticas de ciclo de vida.
3. Elija Crear política de ciclo de vida.
4. Configure las opciones de la política que se describen en los siguientes procedimientos.
5. Para crear la política de ciclo de vida después de configurar las opciones, elija Crear política.

Configuración de políticas: opciones generales

Defina la configuración General de la política.

1. Seleccione la opción AMI en Tipo de política.
2. Escriba el Nombre de la política.
3. Si lo desea, escriba una Descripción para la política de ciclo de vida.
4. De forma predeterminada, la opción Activar está activada. La configuración predeterminada activa la política de ciclo de vida y la agrega a la programación de inmediato. Para crear una política que esté inicialmente desactivada, puede desactivar la opción Activar.
5. Seleccione el rol de IAM que creó para los permisos de la política de ciclo de vida. Si aún no ha creado este rol, consulte Requisitos previos para obtener más información.

Configure el Alcance de la regla para la política.

En esta sección, se configura la selección de recursos para la política de ciclo de vida, en función del tipo de filtro que utilice.

1. Tipo de filtro: recetas: para aplicar las reglas del ciclo de vida a los recursos de imágenes en función de la receta que los creó, seleccione hasta 50 versiones de recetas para la política.
2. Tipo de filtro: etiquetas: para aplicar las reglas del ciclo de vida a los recursos de imágenes en función de las etiquetas de los recursos, ingrese una lista de hasta 50 pares de clave-valor para que coincida con la política.

Regla de eliminación

En el caso de las imágenes de contenedor, esta regla elimina el recurso de imagen de contenedor de Generador de imágenes. Si lo desea, puede eliminar las imágenes de Docker que se distribuyeron en los repositorios de ECR para evitar que se utilicen para ejecutar nuevos contenedores.

Al configurar la regla de eliminación por antigüedad, Generador de imágenes elimina el recurso de imagen tras el periodo de tiempo que haya configurado. Por ejemplo, elimine los recursos de imágenes después de 6 meses. Al configurarla por recuento, Generador de imágenes retiene el número más reciente de imágenes que especifique, o lo más cerca posible a ese número, y elimina las versiones anteriores.

- Por antigüedad
  - Recuento de unidades: especifique el valor entero del periodo de tiempo que debe transcurrir después de crear un recurso de imagen para eliminarlo.
  - Unidad: seleccione el intervalo de tiempo que se va a utilizar. El intervalo puede ser `Days`, `Weeks`, `Months` o `Years`.
  - Retener al menos una imagen: active la casilla para mantener solo el último recurso de imagen disponible para cada versión de la receta a la que afecte esta regla.

  Por recuento
  - Recuento de imágenes: especifique el valor entero del número de recursos de imagen recientes que se deben conservar para cada versión de la receta.
- Eliminar imágenes de contenedor de ECR: seleccione la casilla para eliminar las imágenes de contenedores asociadas almacenadas en un repositorio de ECR. Ya no puede utilizar la imagen de contenedor como base para crear nuevas imágenes ni para ejecutar nuevos contenedores.
- Retener las imágenes con etiquetas asociadas: seleccione la casilla para ingresar una lista de etiquetas para los recursos de imagen que desea conservar.

Etiquetas (opcional)

Agregue etiquetas a la política de ciclo de vida.

AWS CLI

Para crear una nueva política de ciclo de vida de Generador de imágenes, puede utilizar el comando create-lifecycle-policy de la AWS CLI.

Funcionamiento de las reglas de administración del ciclo de vida para recursos de imagen de Generador de imágenes

Las políticas del ciclo de vida de las imágenes utilizan las reglas del ciclo de vida que defina para implementar su estrategia general de administración de recursos. Las reglas que defina ayudan a garantizar la frescura de las imágenes disponibles y a minimizar los costes de la infraestructura subyacente AMIs, como el almacenamiento de instantáneas para la impresión o el almacenamiento en el repositorio de ECR y las tasas de transferencia de datos para las imágenes en contenedores.

Puede configurar los siguientes tipos de reglas para las políticas.

Regla de obsolescencia

Establece el estado del recurso de imagen de Generador de imágenes en `Deprecated`. Las canalizaciones de Generador de imágenes siguen ejecutándose para imágenes obsoletas. Si lo desea, puede establecer el tiempo de obsolescencia de la instancia asociada AMIs sin que ello afecte a su capacidad de lanzar nuevas instancias.

Cuando una AMI queda obsoleta, las búsquedas generales la ignoran. Por ejemplo, si ejecutas el EC2 describe-images comando Amazon en AWS CLI, no volverá a estar obsoleto AMIs en el conjunto de resultados. Sin embargo, todavía puedes encontrarlos en desuso AMIs con su ID de AMI.

Esta regla no está disponible para las imágenes de contenedor.

Regla de deshabilitación

Establece el estado del recurso de imagen de Generador de imágenes en `Disabled`. Esto impide que se ejecuten canalizaciones de Generador de imágenes para esta imagen. Si lo desea, puede deshabilitar la AMI asociada para evitar el lanzamiento de nuevas instancias.

Cuando una AMI está deshabilitada, se convierte en privada y no se puede utilizar para lanzar nuevas instancias. Si ha compartido la AMI con alguna cuenta, organización o unidad organizativa, estas perderán el acceso a la AMI cuando pase a ser privada.

Esta regla no está disponible para las imágenes de contenedor.

Regla de eliminación

Elimina los recursos de imagen por antigüedad o por recuento. Debe definir el umbral que satisfaga sus necesidades. Cuando un recurso de imagen de Generador de imágenes supera el umbral, se elimina. Si lo desea, puede anular el registro de las instantáneas asociadas AMIs o eliminar las instantáneas correspondientes. AMIs También puede especificar etiquetas para los recursos que desea retener más allá del umbral.

En el caso de las imágenes de contenedor, esta regla elimina el recurso de imagen de contenedor de Generador de imágenes. Si lo desea, puede eliminar las imágenes de contenedor que se distribuyeron en los repositorios de ECR para evitar que se utilicen para ejecutar nuevos contenedores.

Contenido

- Reglas del ciclo de vida de las AMI
- Visualización de los detalles de las reglas de administración del ciclo de vida de una política

Reglas del ciclo de vida de las AMI

Las siguientes reglas de exclusión definen las excepciones a las reglas del ciclo de vida para. AMIs AMIs las que cumplen los criterios especificados en las reglas de exclusión se excluyen de las acciones del ciclo de vida. Las reglas de exclusión no están disponibles en la Consola de administración de AWS.

Los siguientes términos utilizan la notación de API del tipo de datos `LifecyclePolicyDetailExclusionRules`.

Reglas de exclusión

amis

Contiene la configuración de `LifecyclePolicyDetailExclusionRulesAmis` que aparecen en la lista que sigue.

tagMap

Puede proporcionar una lista de hasta 50 etiquetas que omitan las acciones del ciclo de vida de cualquier tipo de recurso.

Los siguientes términos utilizan la notación de API del tipo de datos `LifecyclePolicyDetailExclusionRulesAmis`.

Reglas de exclusión de AMI

isPublic

Configura si el público AMIs queda excluido de la acción del ciclo de vida.

lastLaunched

Especifica los detalles de configuración de Generador de imágenes para excluir los recursos más recientes de las acciones del ciclo de vida.

regiones

Configura los elementos Regiones de AWS excluidos de la acción del ciclo de vida.

sharedAccounts

Especifica Cuentas de AWS qué recursos se excluyen de la acción del ciclo de vida.

tagMap

Enumera las etiquetas que deben excluirse de las acciones del ciclo de vida para las AMIs que las tienen.

Visualización de los detalles de las reglas de administración del ciclo de vida de una política

Las reglas se definen en las políticas de administración del ciclo de vida que se crean para los recursos de imágenes de Generador de imágenes. En la consola, la página de detalles de la política del ciclo de vida contiene una Pestaña Reglas en la que se muestran los detalles de las reglas que configuró para la política.

Para obtener los detalles de la política en AWS CLI, puede ejecutar el get-lifecycle-policycomando. Los detalles de la política en la respuesta contienen una lista de las acciones (reglas) que ha definido para la política, que incluyen todas las opciones configuradas.

Configuración de imágenes personalizadas con Generador de imágenes

Los recursos de configuración son los componentes que conforman las canalizaciones de imágenes, así como las imágenes que producen esas canalizaciones. En este capítulo se explica la creación, mantenimiento y uso compartido de los recursos de Generador de Imágenes, incluidos los componentes, las recetas y las imágenes, junto con la configuración de infraestructura y de distribución.

> **Note**
>
> Para ayudar a administrar los recursos de Generador de Imágenes, puede asignar sus propios metadatos a cada recurso en forma de etiquetas. Se utilizan etiquetas para clasificar los recursos de AWS de diversas maneras, por ejemplo, según su finalidad, propietario o entorno. Esto es útil cuando se tienen muchos recursos del mismo tipo. Puede identificar más fácilmente un recurso específico según las etiquetas que le haya asignado.
> Para obtener más información sobre cómo etiquetar los recursos mediante los comandos de Image Builder de AWS CLI, consulte la Etiquetar recursos sección de esta guía.

Contenido

- Administración de recetas en Generador de imágenes
- Administre la configuración de la infraestructura de Generador de imágenes
- Administrar los ajustes de la distribución de Generador de imágenes

Administración de recetas en Generador de imágenes

Una receta de EC2 Image Builder define la imagen base que se utilizará como punto de partida para crear una nueva imagen, junto con el conjunto de componentes que se añaden para personalizar la imagen y comprobar que todo funciona como se espera. Generador de Imágenes proporciona opciones de versiones automáticas para cada componente. De forma predeterminada, puede aplicar hasta 20 componentes a una receta. Esto incluye tanto los componentes de creación como los de prueba.

Después de crear una receta, no puede modificarla ni reemplazarla. Para actualizar los componentes después de crear una receta, debe crear una nueva receta o versión de la receta. Siempre puede aplicar etiquetas a sus recetas existentes. Para obtener más información sobre cómo etiquetar los recursos mediante los comandos de Image Builder de AWS CLI, consulte la Etiquetar recursos sección de esta guía.

> **Tip**
>
> Puedes usar componentes gestionados por Amazon en tus recetas o puedes desarrollar tus propios componentes personalizados. Para obtener más información, consulte Desarrollo de componentes personalizados para su imagen de Generador de imágenes. Para las recetas de imágenes que generan resultados AMIs, también puedes usar productos y componentes de AWS Marketplace imágenes. Para obtener más información sobre la integración con AWS Marketplace los productos, consulteAWS Marketplace integración en Image Builder.

En esta sección se explica cómo enumerar, ver y crear recetas.

Contenido

- Enumeración y visualización de los detalles de una receta de imágenes
- Enumeración y visualización de los detalles de una receta de contenedor
- Creación de una nueva versión de una receta de imagen
- Creación de una nueva versión de una receta de contenedor
- Eliminar recursos

Enumeración y visualización de los detalles de una receta de imágenes

En esta sección se describen las distintas formas en las que puede encontrar información y ver los detalles de sus recetas de imágenes de EC2 Image Builder.

Detalles de receta de imágenes

- Enumerar recetas de imágenes de la consola
- Enumere las recetas con imágenes del AWS CLI
- Ver los detalles de la receta de imágenes de la consola
- Obtenga los detalles de las recetas en imágenes del AWS CLI

- Obtenga información detallada sobre la política de recetas con imágenes en el AWS CLI

Enumerar recetas de imágenes de la consola

Para ver una lista de las recetas de imágenes creadas en su cuenta en la consola de Generador de Imágenes, siga estos pasos:

1. Abra la consola EC2 de Image Builder en https://console.aws.amazon.com/imagebuilder/.
2. Elija Recetas de imágenes en el panel de navegación. Aquí se muestra una lista de las recetas de imágenes que se han creado en su cuenta.
3. Para ver los detalles o crear una nueva versión de la receta, seleccione el enlace del nombre de la receta. Esto abre la vista detallada de la receta.

> **Note**
>
> También puede seleccionar la casilla junto al nombre de la receta, y, a continuación, elegir Ver detalles.

Enumere las recetas con imágenes del AWS CLI

En el siguiente ejemplo se indica cómo enumerar todas las recetas de imágenes mediante la AWS CLI.

```
aws imagebuilder list-image-recipes
```

Ver los detalles de la receta de imágenes de la consola

Para ver los detalles de una receta de imágenes específica usando la consola de Generador de Imágenes, seleccione la receta de imágenes que desee revisar siguiendo los pasos que se describen en Enumerar recetas de imágenes de la consola.

En la página de detalles de la receta, puede hacer lo siguiente:

- Elimine la receta. Para obtener más información sobre cómo eliminar recursos en Generador de Imágenes, consulte Eliminación de recursos de Generador de imágenes obsoletos o no utilizados.
- Cree una nueva versión.

- Cree una canalización a partir de la receta. Después de seleccionar Crear canalización a partir de esta receta, accederá al asistente de canalización. Para obtener más información sobre la creación de una canalización de Generador de Imágenes usando el asistente de canalización, consulte Tutorial: creación de una canalización de imágenes con una AMI de salida desde el asistente de la consola de Generador de Imágenes

> **Note**
>
> Al crear una canalización a partir de una receta existente, la opción de crear una nueva receta no está disponible.

Obtenga los detalles de las recetas en imágenes del AWS CLI

El siguiente ejemplo muestra cómo usar un comando imagebuilder de la CLI para ver los detalles de una receta de imágenes especificando su nombre de recurso de Amazon (ARN).

```
aws imagebuilder get-image-recipe --image-recipe-arn arn:aws:imagebuilder:us-
west-2:123456789012:image-recipe/my-example-recipe/2020.12.03
```

Obtenga información detallada sobre la política de recetas con imágenes en el AWS CLI

El siguiente ejemplo muestra cómo utilizar un comando imagebuilder de la CLI para ver los detalles de una política de receta de imágenes especificando su ARN.

```
aws imagebuilder get-image-recipe-policy --image-recipe-arn arn:aws:imagebuilder:us-
west-2:123456789012:image-recipe/my-example-recipe/2020.12.03
```

Enumeración y visualización de los detalles de una receta de contenedor

En esta sección se describen las formas en las que puede encontrar información y ver los detalles de sus recetas de contenedores de EC2 Image Builder.

Detalles de la receta de contenedor

- Enumeración de las recetas de contenedor en la consola
- Enumere las recetas de contenedor con la AWS CLI

- Ver los detalles de la receta de contenedor en la consola
- Obtención de los detalles de la receta de contenedor con la AWS CLI
- Obtenga los detalles de la política de recetas de envases con el AWS CLI

Enumeración de las recetas de contenedor en la consola

Para ver una lista de las recetas de contenedor que se han creado en su cuenta en la consola de Generador de Imágenes, siga estos pasos:

1. Abra la consola EC2 de Image Builder en https://console.aws.amazon.com/imagebuilder/.
2. En el panel de navegación, elija Recetas de contenedor. Aquí se muestra una lista de las recetas de contenedor que se han creado en su cuenta.
3. Para ver los detalles o crear una nueva versión de la receta, seleccione el enlace del nombre de la receta. Esto abre la vista detallada de la receta.

> **Note**
>
> También puede seleccionar la casilla junto al nombre de la receta, y, a continuación, elegir Ver detalles.

Enumere las recetas de contenedor con la AWS CLI

En el ejemplo siguiente se indica cómo enumerar todas sus recetas de contenedor mediante la AWS CLI.

```
aws imagebuilder list-container-recipes
```

Ver los detalles de la receta de contenedor en la consola

Para ver los detalles de una receta de contenedor específica con la consola de Generador de Imágenes, seleccione la receta de contenedor que desee revisar y siga los pasos que se describen en Enumeración de las recetas de contenedor en la consola.

En la página de detalles de la receta, puede hacer lo siguiente:

- Elimine la receta. Para obtener más información sobre cómo eliminar recursos en Generador de Imágenes, consulte Eliminación de recursos de Generador de imágenes obsoletos o no utilizados.

- Cree una nueva versión.
- Cree una canalización a partir de la receta. Tras seleccionar Crear canalización a partir de esta receta, accederá al asistente de canalización. Para obtener más información sobre cómo crear una canalización de Generador de Imágenes mediante el asistente de canalización, consulte Tutorial: creación de una canalización de imágenes con una AMI de salida desde el asistente de la consola de Generador de Imágenes.

> **Note**
>
> Al crear una canalización a partir de una receta existente, la opción de crear una receta nueva no está disponible.

Obtención de los detalles de la receta de contenedor con la AWS CLI

El siguiente ejemplo muestra cómo utilizar un comando imagebuilder CLI de para obtener los detalles de una receta de contenedor especificando su ARN.

```
aws imagebuilder get-container-recipe --container-recipe-arn arn:aws:imagebuilder:us-
west-2:123456789012:container-recipe/my-example-recipe/2020.12.03
```

Obtenga los detalles de la política de recetas de envases con el AWS CLI

El siguiente ejemplo muestra cómo utilizar un comando imagebuilder de la CLI de para obtener los detalles de la política de una receta de contenedor especificando su ARN.

```
aws imagebuilder get-container-recipe-policy --container-recipe-arn
 arn:aws:imagebuilder:us-west-2:123456789012:container-recipe/my-example-
recipe/2020.12.03
```

Creación de una nueva versión de una receta de imagen

En esta sección se describe cómo crear una nueva versión de una receta de imagen.

Contenido

- Creación de una nueva versión de una receta de imagen de la consola
- Cree una receta con imágenes con el AWS CLI
- Importación de una máquina virtual como imagen base en la consola

Creación de una nueva versión de una receta de imagen de la consola

Cuando crea una nueva versión de la receta, es prácticamente lo mismo que crear una nueva receta. La diferencia es que, en la mayoría de los casos, ciertos detalles están preseleccionados para que coincidan con la receta base. La siguiente lista describe las diferencias entre crear una receta nueva y crear una nueva versión de una receta existente.

Los detalles básicos de la receta en la nueva versión

- Nombre: no se puede editar.
- versión: obligatoria. Introduzca el número de versión que desee crear en el formato <major>.<minor>.<patch>. Image Builder admite el incremento automático de versiones de las recetas, lo que le permite utilizar patrones comodín en sus versiones de recetas. Al crear una receta con una versión comodín, por ejemplo`1.0.x`, Image Builder incrementa automáticamente la versión (por ejemplo,`1.0.1`,`1.0.2`,`1.0.3`, etc.). Esto elimina la necesidad de realizar un seguimiento e incrementar manualmente las versiones de la receta.
- La opción Seleccionar imagen: está preseleccionada, pero puede editarla. Si cambia la elección para la fuente de su imagen base, es posible que pierda otros detalles que dependen de la opción original que haya elegido.

  Para ver los detalles asociados con la selección de su imagen base, elija la pestaña que coincida con su selección.

  Managed image

  - Sistema operativo (SO) de la imagen: no se puede editar.
  - Nombre de la imagen: preseleccionada, en función de la combinación de imágenes base que haya elegido para la receta existente. Sin embargo, si cambia la opción Seleccionar imagen, se pierde el nombre de la imagen preseleccionada.
  - Opciones de control de versiones automático: no coincide con la receta base. Esta opción de imagen utiliza por defecto la opción Usar la versión del SO seleccionada.

    > **Important**
    >
    > Si utiliza el control de versiones semántico para iniciar las compilaciones de canalización, asegúrese de cambiar este valor a Usar la versión más reciente disponible del SO. Para obtener más información sobre el control de versiones

semántico para los recursos de Generador de Imágenes, consulte Control de versiones semántico en Generador de imágenes.

AWS Marketplace image

- Suscripciones: esta pestaña debe estar abierta y la imagen de la que se ha suscrito AWS Marketplace debe estar preseleccionada para que coincida con la receta base. Si cambia la imagen que su receta utiliza como imagen base, es posible que pierda otros detalles que dependen de la imagen original que eligió.

Para obtener más información sobre AWS Marketplace los productos, consulta Comprar productos en la Guía del AWS Marketplace comprador.

Custom AMI

Origen de la AMI (obligatorio): introduzca el ID de la AMI o un parámetro del almacén de parámetros AWS Systems Manager (SSM) que contenga un ID de AMI para utilizarlo como imagen base. El agente SSM debe estar preinstalado en la AMI seleccionada.

- ID de AMI: esta configuración no viene rellenada previamente con la entrada original. Introduzca el ID de AMI de la imagen base. Ejemplo: *ami-1234567890abcdef1*.
- Parámetro SSM: introduzca el nombre o el ARN del parámetro SSM Parameter Store que contiene el ID de AMI de la imagen base. Ejemplo: */ib/test/param* o `arn:aws:ssm:`*us-east-1*`:`*111122223333*`:parameter`*/ib/test/param*.

- Configuración de la instancia: los ajustes están preseleccionados, pero puede editarlos.
  - Systems Manager Agent: puede activar o desactivar esta casilla de verificación para controlar la instalación del agente de Systems Manager en la nueva imagen. La casilla de verificación está desactivada de forma predeterminada para incluir Systems Manager Agent en su nueva imagen. Para eliminar Systems Manager Agent de la imagen final, active la casilla de verificación para que el agente no se incluya en la AMI.
  - Datos de usuario: utilice esta área para proporcionar comandos o un script de comandos para que se ejecuten al lanzar su instancia de compilación. Sin embargo, este valor sustituye a cualquier comando que Generador de Imágenes pudiera haber añadido para asegurarse de que Systems Manager se instale. Estos comandos incluyen el script de limpieza que Generador de Imágenes normalmente ejecuta para las imágenes de Linux antes de crear la nueva imagen.

    Cuando Image Builder lanza una instancia, los scripts de datos de usuario se ejecutan durante la fase de inicio de la nube, antes de que comience la ejecución de los componentes. Este paso se registra en el siguiente archivo de la instancia:. `var/log/cloud-init.log`

> **Note**
>
> - Si introduce datos de usuario, asegúrese de que Systems Manager Agent esté preinstalado en su imagen base o de incluir la instalación en sus datos de usuario.
> - En el caso de las imágenes de Linux, asegúrese de ejecutar los pasos de limpieza incluyendo un comando para crear un archivo vacío denominado `perform_cleanup` en el script de sus datos de usuario. Generador de Imágenes detecta este archivo y ejecuta el script de limpieza antes de crear la nueva imagen. Para obtener más información y un script de muestra, consulte Prácticas recomendadas de seguridad para Generador de imágenes.

- Directorio de trabajo: está preseleccionado, pero se puede editar.
- Componentes: los componentes que ya están incluidos en la receta se muestran en la sección Componentes seleccionados al final de cada una de las listas de componentes (compilación y prueba). Puede eliminar o reordenar los componentes seleccionados para adaptarlos a sus necesidades.

  Los componentes de endurecimiento del CIS no siguen las reglas de ordenación de componentes estándar de las recetas de Generador de Imágenes. Los componentes de endurecimiento del CIS siempre se ejecutan en último lugar para garantizar que las pruebas de referencia se ejecuten con la imagen de salida.

  > **Note**
  >
  > Las listas de componentes de compilación y prueba muestran los componentes disponibles según el tipo de propietario del componente. Para agregar un componente, elija Agregar componentes de compilación y seleccione el filtro de propiedad que corresponda. Por ejemplo, para añadir un componente de compilación asociado a un AWS Marketplace producto, seleccione`AWS Marketplace`. Esto abre un panel de selección en el lado derecho de la interfaz de la consola que muestra AWS Marketplace los componentes.
  > Para el componente CIS, seleccione`Third party managed`.

  Puede configurar las siguientes opciones para su componente seleccionado:

- Opciones de control de versiones: preseleccionadas, pero se pueden cambiar. Le recomendamos que elija la opción Usar la última versión del componente disponible para asegurarse de que las compilaciones de imágenes siempre incluyan la última versión del componente. Si necesita usar una versión de componente específica en su receta, puede elegir Especificar la versión del componente e introducir la versión en el cuadro versión del componente que aparece.
- Parámetros de entrada: muestra los parámetros de entrada que acepta el componente. El valor está precargado con el valor de la versión anterior de la receta. Si utiliza este componente por primera vez en esta receta y se definió un valor predeterminado para el parámetro de entrada, el valor predeterminado aparece en el cuadro Valor con el texto atenuado. Si no se especifica ningún otro valor, Generador de imágenes utiliza el valor predeterminado.

  Si se requiere un parámetro de entrada, pero no tiene un valor predeterminado definido en el componente, debe proporcionar un valor. Generador de imágenes no creará la versión de la receta si falta algún parámetro obligatorio y no tiene definido un valor predeterminado.

  > **Important**
  >
  > Los parámetros del componente son valores de texto sin formato y se registran en AWS CloudTrail. Le recomendamos que utilice AWS Secrets Manager el almacén de AWS Systems Manager parámetros para almacenar sus secretos. Para obtener más información sobre Secrets Manager, consulte ¿Qué es Secrets Manager? en la Guía del usuario de AWS Secrets Manager . Para obtener más información acerca del almacén de parámetros de AWS Systems Manager , consulte Almacén de parámetros de AWS Systems Manager en la Guía del usuario de AWS Systems Manager .

  Para ampliar la configuración de las opciones de control de versiones o los parámetros de entrada, puede seleccionar la flecha junto al nombre de la configuración. Para ampliar todos los ajustes de todos los componentes seleccionados, puede activar y desactivar el interruptor Expandir todo.

- Almacenamiento (volúmenes): vienen precargados. Las selecciones de nombre de dispositivo, Instantánea e IOPS del volumen raíz no se pueden editar. Sin embargo, puede cambiar todos los ajustes restantes, como el tamaño. También puede agregar nuevos volúmenes y cifrar los volúmenes nuevos o existentes.

  Para cifrar los volúmenes de las imágenes que Generador de Imágenes crea en su cuenta en la región de origen (donde se ejecuta la compilación), debe utilizar el cifrado del volumen de

almacenamiento de la receta de la imagen. El cifrado que se ejecuta durante la fase de distribución de la creación es solo para las imágenes que se distribuyen a otras cuentas o regiones.

> **Note**
>
> Si utiliza el cifrado para sus volúmenes, debe seleccionar la clave para cada volumen por separado, incluso si la clave es la misma que se utiliza para el volumen raíz.

Para crear una nueva versión de una receta de imagen:

1. En la parte superior de la página de detalles de la receta, seleccione Crear una nueva versión. Esto le llevará a la página Crear receta de imagen.
2. Para crear la nueva versión, realice los cambios y, a continuación, elija Crear receta.

   La imagen final puede contener hasta cuatro códigos de producto de productos y componentes de la AWS Marketplace imagen. Si la imagen base y los componentes seleccionados contienen más de cuatro códigos de producto, Image Builder devolverá un error al intentar crear la receta.

Para obtener más información sobre cómo crear una receta de imagen al crear una canalización de imágenes, consulte Paso 2: elegir receta en la sección Introducción de esta guía.

Cree una receta con imágenes con el AWS CLI

Para crear una receta de imagen con el `create-image-recipe` comando Image Builder de AWS CLI, siga estos pasos:

Requisitos previos

Antes de ejecutar los comandos de Image Builder de esta sección para crear una receta de imagen a partir de AWS CLI, si lo desea, puede crear los componentes que utilice la receta. El ejemplo de receta de imagen que se muestra en el siguiente paso hace referencia a los componentes de ejemplo que se crean en la sección Cree un componente personalizado a partir del AWS CLI de esta guía.

Si desea incluir componentes en la receta, anote los ARNs que desee incluir. También puede crear recetas sin ningún componente para probar los flujos de trabajo existentes AMIs o de distribución exclusiva.

1. Creación de un archivo JSON de entrada de la CLI

   Puede proporcionar todas las entradas para el comando create-image-recipe con parámetros de comando en línea. Sin embargo, el comando resultante puede ser bastante largo. Para simplificar el comando, puede proporcionar en su lugar un archivo JSON que contenga todas las configuraciones de la receta.

   > **Note**
   >
   > La convención de nomenclatura de los valores de datos del archivo JSON sigue el patrón que se especifica para los parámetros de solicitud de operación de la API Image Builder. Para revisar los parámetros de la solicitud de operación de la API, consulte el CreateImageRecipecomando en la Referencia de la API de EC2 Image Builder.
   > Para proporcionar los valores de los datos como parámetros de la línea de comandos, consulte los nombres de los parámetros especificados en la referencia de comandos de la AWS CLI .

   Este es un resumen de los parámetros que especificamos en este ejemplo:

   - Nombre (cadena, obligatorio): nombre de la receta de imagen.
   - Descripción (cadena): descripción de la receta de imagen.
   - parentImage (cadena, obligatorio): imagen que la receta de imagen utiliza como base para su imagen personalizada. Puede especificar la imagen principal mediante una de las siguientes opciones:
     - ID DE AMI
     - Recurso de imágenes de Image Builder (ARN)
     - AWS Systems Manager (SSM) Parámetro del almacén de parámetros, con el prefijo `ssm:` y seguido del nombre del parámetro o ARN.
     - AWS Marketplace ID del producto

     > **Note**
     >
     > Los ejemplos de Linux y macOS utilizan una AMI de Image Builder y el ejemplo de Windows utiliza un ARN.

- SemanticVersion <major>(cadena, obligatorio): introduzca el número de versión que desea crear en el formato. <minor>. <patch>. Image Builder admite el incremento automático de versiones de las recetas, lo que le permite utilizar patrones comodín en sus versiones de recetas. Al crear una receta con una versión comodín, por ejemplo`1.0.x`, Image Builder incrementa automáticamente la versión (por ejemplo,`1.0.1`,`1.0.2`,`1.0.3`, etc.). Esto elimina la necesidad de realizar un seguimiento e incrementar manualmente las versiones de la receta. Para obtener más información sobre el control de versiones semántico para los recursos de Generador de Imágenes, consulte Control de versiones semántico en Generador de imágenes.
- componentes (matriz, opcional): contiene una matriz de `ComponentConfiguration` objetos. Los componentes son opcionales: puede crear recetas sin ningún componente para los flujos de trabajo de prueba o distribución:

  > **Note**
  >
  > Generador de Imágenes instala los componentes en el orden en que se especificaron en la receta. Sin embargo, los componentes de refuerzo del CIS siempre se ejecutan en último lugar para garantizar que las pruebas de referencia se ejecutan con su imagen de salida.

  - componentARN (cadena, obligatorio): el ARN del componente.

    > **Tip**
    >
    > Para usar uno de los ejemplos para crear su propia receta ARNs con imágenes, debe reemplazar el ejemplo ARNs por el de los componentes que está utilizando para su receta.

  - Parámetros (matriz de objetos): contiene una matriz de objetos `ComponentParameter`. Si se requiere un parámetro de entrada, pero no tiene un valor predeterminado definido en el componente, debe proporcionar un valor. Generador de imágenes no creará la versión de la receta si falta algún parámetro obligatorio y no tiene definido un valor predeterminado.

> **Important**
>
> Los parámetros del componente son valores de texto sin formato y se registran en AWS CloudTrail. Le recomendamos que utilice AWS Secrets Manager o el almacén de AWS Systems Manager parámetros para almacenar sus secretos. Para obtener más información sobre Secrets Manager, consulte ¿Qué es Secrets Manager? en la Guía del usuario de AWS Secrets Manager . Para obtener más información acerca del almacén de parámetros de AWS Systems Manager , consulte Almacén de parámetros de AWS Systems Manager en la Guía del usuario de AWS Systems Manager .

    - Nombre (cadena, obligatorio): el nombre del parámetro del componente que se va a establecer.
    - valor (matriz de cadenas, obligatorio): contiene una matriz de cadenas para establecer el valor del parámetro de componente nombrado. Si hay un valor por defecto definido para el componente y no se proporciona ningún otro valor, TOE de AWS utiliza el valor por defecto.
- additionalInstanceConfiguration(objeto): especifique ajustes adicionales y ejecute scripts para las instancias de compilación.
    - systemsManagerAgent(objeto): contiene la configuración del agente de Systems Manager en la instancia de compilación.
        - uninstallAfterBuild(Boolean): controla si el agente de Systems Manager se elimina de la imagen de compilación final antes de crear la nueva AMI. Si esta opción se establece como `true`, el agente se elimina de la imagen final. Si la opción se establece como `false`, entonces se deja al agente activo para que se incluya en la nueva AMI. El valor predeterminado es `false`.

> **Note**
>
> Si el atributo `uninstallAfterBuild` no está incluido en el archivo JSON y se cumplen las siguientes condiciones, Generador de Imágenes elimina Systems Manager Agent de la imagen final para que no esté disponible en la AMI:
>
> - El `userDataOverride` está vacío o se ha omitido del archivo JSON.

- Generador de Imágenes instaló Systems Manager Agent automáticamente en la instancia de compilación de un sistema operativo que no tenía el agente preinstalado en la imagen base.

- userDataOverride(cadena): proporciona comandos o un script de comandos para que se ejecuten al lanzar la instancia de compilación.

> **Note**
>
> Los datos de usuario siempre están codificados en base 64.
> Por ejemplo, los siguientes comandos se codifican como `IyEvYmluL2Jhc2gKbWtkaXIgLXAgL3Zhci9iYi8KdG91Y2ggL3Zhcg==`:
>
> ```
> #!/bin/bash
> mkdir -p /var/bb/
> touch /var
> ```
>
> El ejemplo de Linux utiliza este valor codificado.

Linux

La imagen base (propiedad de `parentImage`) del siguiente ejemplo es una AMI. Cuando utilice una AMI, debe tener acceso a la AMI y la AMI debe estar en la región de origen (la misma región en la que Generador de Imágenes ejecuta el comando). Guarde el archivo como `create-image-recipe.json` y utilícelo en el comando create-image-recipe.

```
{
"name": "BB Ubuntu Image recipe",
"description": "Hello World image recipe for Linux.",
"parentImage": "ami-1234567890abcdef1",
"semanticVersion": "1.0.0",
"components": [
 {
  "componentArn": "arn:aws:imagebuilder:us-west-2:111122223333:component/bb$"
 }
],
"additionalInstanceConfiguration": {
 "systemsManagerAgent": {
   "uninstallAfterBuild": true
```

```
 },
 "userDataOverride": "IyEvYmluL2Jhc2gKbWtkaXIgLXAgL3Zhci9iYi8KdG91Y2ggL3Zhcg=="
}
}
```

Windows

El siguiente ejemplo hace referencia a la última versión de la imagen base completa en inglés de Windows Server 2016. El ARN de este ejemplo hace referencia a la imagen más reciente en función de los filtros de versión semántica que especificó:. `arn:aws:imagebuilder:us-west-2:aws:image/windows-server-2016-english-full-base-x86/x.x.x`

```
{
"name": "MyBasicRecipe",
"description": "This example image recipe creates a Windows 2016 image.",
"parentImage": "arn:aws:imagebuilder:us-west-2:aws:image/windows-server-2016-english-full-base-x86/x.x.x",
"semanticVersion": "1.0.0",
"components": [
 {
  "componentArn": "arn:aws:imagebuilder:us-west-2:111122223333:component/my-example-component/2019.12.02/1"
 },
 {
  "componentArn": "arn:aws:imagebuilder:us-west-2:111122223333:component/my-imported-component/1.0.0/1"
 }
]
}
```

> **Note**
>
> Para obtener más información sobre el control de versiones semántico para los recursos de Generador de Imágenes, consulte Control de versiones semántico en Generador de imágenes.

macOS

La imagen base (propiedad de `parentImage`) del siguiente ejemplo es una AMI. Cuando utilice una AMI, debe tener acceso a la AMI y la AMI debe estar en la región de origen (la misma región en la que Generador de Imágenes ejecuta el comando). Guarde el archivo como `create-image-recipe.json` y utilícelo en el comando create-image-recipe.

```
{
"name": "macOS Catalina Image recipe",
"description": "Hello World image recipe for macOS.",
"parentImage": "ami-1234567890abcdef1",
"semanticVersion": "1.0.0",
"components": [
 {
  "componentArn": "arn:aws:imagebuilder:us-
west-2:111122223333:component/catalina$"
 }
],
"additionalInstanceConfiguration": {
 "systemsManagerAgent": {
   "uninstallAfterBuild": true
 },
 "userDataOverride": "IyEvYmluL2Jhc2gKbWtkaXIgLXAgL3Zhci9iYi8KdG91Y2ggL3Zhcg=="
}
}
```

Ejemplo: Receta sin componentes

Puede crear recetas sin ningún componente para probar los flujos de trabajo existentes AMIs o de distribución exclusiva. El siguiente ejemplo muestra una receta que usa una AMI existente sin aplicar ningún componente adicional:

```
{
 "name": "Test Distribution Recipe",
 "description": "Recipe for testing and distributing existing AMI without
 modifications.",
 "parentImage": "ami-1234567890abcdef1",
 "semanticVersion": "1.0.0",
 "additionalInstanceConfiguration": {
```

```
  "systemsManagerAgent": {
    "uninstallAfterBuild": true
  }
 }
}
```

2. Creación de la receta

   Utilice el siguiente comando para crear la receta. Introduzca el nombre del archivo JSON que creó en el paso anterior en el parámetro `--cli-input-json`:

   ```
   aws imagebuilder create-image-recipe --cli-input-json file://create-image-recipe.json
   ```

   > **Note**
   >
   > - Debe incluir la notación `file://` al principio de la ruta del archivo JSON.
   > - La ruta del archivo JSON debe seguir la convención apropiada para el sistema operativo base donde se está ejecutando el comando. Por ejemplo, Windows utiliza la barra diagonal inversa (\) para referirse a la ruta de directorios, mientras que Linux y macOS usan la barra diagonal (/).

   La imagen final puede contener hasta cuatro códigos de producto de productos y componentes de la AWS Marketplace imagen. Si la imagen base y los componentes seleccionados contienen más de cuatro códigos de producto, Image Builder devuelve un error al ejecutar el `create-image-recipe` comando.

Importación de una máquina virtual como imagen base en la consola

En esta sección, nos centraremos en cómo importar una máquina virtual (VM) como imagen base para su receta de imagen. Aquí no abordamos otros pasos relacionados con la creación de una receta o una versión de la receta. Para ver pasos adicionales para crear una nueva receta de imagen con el asistente de creación de canalizaciones de la consola de Generador de Imágenes, consulte Asistente de canalización: crear AMI. Para ver pasos adicionales para crear una nueva receta de imagen o versión de una receta, consulte Creación de una nueva versión de una receta de imagen.

Para importar una máquina virtual como imagen base para su receta de imagen en la consola de Generador de Imágenes, siga estos pasos, junto con cualquier otro paso necesario, para crear su receta o versión de la receta.

1. En la sección Seleccionar imagen para la imagen base, seleccione la opción Importar imagen base.
2. Elija el sistema operativo (SO) de la imagen y la versión del SO como lo haría normalmente.

Configuración de la importación de máquinas virtuales

Cuando exporta su máquina virtual desde su entorno de virtualización, ese proceso crea un conjunto de uno o más archivos de contenedor de disco que actúan como instantáneas del entorno, la configuración y los datos de su máquina virtual. Puede utilizar estos archivos para importar su máquina virtual como imagen base para su receta de imagen. Para obtener más información sobre la importación VMs en Image Builder, consulte Importación y exportación de imágenes de máquinas virtuales

Para especificar la ubicación del origen de importación, siga estos pasos:

Importación de fuente

Especifique la fuente del primer contenedor de discos o instantáneas de imágenes de máquina virtual que se va a importar en la sección Contenedor de discos 1.

1. Origen: puede ser un bucket de S3 o una instantánea de EBS.
2. Seleccione la ubicación S3 del disco: introduzca la ubicación en Amazon S3 en la que se almacenan las imágenes del disco. Para buscar la ubicación, elija Examinar S3.
3. Para agregar un contenedor de discos, elija Agregar contenedor de disco.

rol de IAM

Para asociar un rol de IAM a su configuración de VM Import, seleccione el rol de la lista desplegable del rol de IAM o elija Crear un nuevo rol para crear uno nuevo. Si crea un rol nuevo, la página de la consola de roles de IAM se abre en una pestaña independiente.

Configuración avanzada: opcional

Los siguientes ajustes son opcionales. Con estos ajustes, puede configurar el cifrado, las licencias, las etiquetas y mucho más para la imagen base que se crea con la importación.

General

1. Especifique un nombre único para la imagen base. Si no introduce ningún valor, la imagen base hereda el nombre de la receta.
2. Especifique una versión para la imagen base. Use el siguiente formato: *<major>.<minor>.<patch>*. Si no introduce un valor, la imagen base hereda la versión de la receta.
3. También puede introducir una descripción para la imagen base.

Arquitectura de la imagen base

Para especificar la arquitectura del origen de importación de la máquina virtual, seleccione un valor de la lista de arquitecturas.

Cifrado

Si las imágenes de disco de la máquina virtual están cifradas, debe proporcionar una clave para utilizarla en el proceso de importación. Para especificar un valor AWS KMS key para la importación, seleccione un valor de la lista de cifrado (clave KMS). La lista contiene las claves KMS a las que tiene acceso su cuenta en la región actual.

Administración de licencias

Al importar una máquina virtual, el proceso de importación detecta automáticamente el SO de la máquina virtual y aplica la licencia adecuada a la imagen base. Según la plataforma del sistema operativo, los tipos de licencia son los siguientes:

- Licencia incluida: se aplica a la imagen base una licencia de AWS adecuada para su plataforma.
- Traiga su propia licencia (BYOL): conserva la licencia de su máquina virtual, si corresponde.

Para adjuntar las configuraciones de licencia creadas con AWS License Manager la imagen base, seleccione un nombre de la configuración de licencia de la lista. Para obtener más información acerca de License Manager, consulte Trabajar con AWS License Manager

> **Note**
>
> - Las configuraciones de la licencia contienen reglas de asignación de licencias que se basan en las condiciones de los contratos de su empresa.

- Linux solo admite licencias BYOL.

Etiquetas (imagen base)

Las etiquetas utilizan pares clave-valor para asignar texto con capacidad de búsqueda a su recurso de Generador de Imágenes. Para especificar etiquetas para la imagen base importada, introduzca los pares clave-valor con los cuadros Clave y Valor.

Para agregar una etiqueta, elija Add tag (Añadir etiqueta). Para quitar una etiqueta, elija Remove tag (Eliminar etiqueta).

Creación de una nueva versión de una receta de contenedor

Esta sección muestra cómo crear una nueva versión de una receta de contenedor.

Contenido

- Creación de una nueva versión de una receta de contenedor con la consola
- Crear una receta de contenedor con la AWS CLI

Creación de una nueva versión de una receta de contenedor con la consola

Crear una nueva versión de una receta de contenedor es prácticamente lo mismo que crear una nueva receta. La diferencia es que, en la mayoría de los casos, ciertos detalles están preseleccionados para que coincidan con la receta base. La siguiente lista describe las diferencias entre crear una receta nueva y crear una nueva versión de una receta existente.

Detalles de la receta

- Nombre: no se puede editar.
- versión: obligatoria. Este detalle no viene precargado con la versión actual ni con ningún tipo de secuencia. Ingrese el número de versión que desee crear en el formato major.minor.patch. Si la versión ya existe, se produce un error.

Imagen base

- Opción Seleccionar imagen: preseleccionada, pero editable. Si cambia la elección para la fuente de su imagen base, es posible que pierda otros detalles que dependen de la opción original que haya elegido.

  En el caso de las imágenes de contenedores de Docker, puede elegir entre imágenes públicas alojadas en DockerHub, imágenes de contenedores existentes en Amazon ECR o imágenes de contenedores gestionados por Amazon. Para ver los detalles asociados con la selección de su imagen base, elija la pestaña que coincida con su selección.

  Managed images

  - Sistema operativo (SO) de la imagen: no se puede editar.
  - Nombre de la imagen: preseleccionada, en función de la combinación de imágenes base que haya elegido para la receta existente. Sin embargo, si cambia la opción Seleccionar imagen, se pierde el nombre de la imagen preseleccionada.
  - Opciones de control de versiones automático: no coincide con la receta base. Las opciones de control de versiones automático tienen de forma predeterminada la opción Usar la versión del sistema operativo seleccionado.

    > **Important**
    >
    > Si utiliza el control de versiones semántico para iniciar las compilaciones de canalización, asegúrese de cambiar este valor a Usar la versión más reciente disponible del SO. Para obtener más información sobre el control de versiones semántico para los recursos de Generador de Imágenes, consulte Control de versiones semántico en Generador de imágenes.

  ECR image

  - Sistema operativo (SO) de la imagen: preseleccionado, pero editable.
  - versión del sistema operativo: preseleccionado, pero editable.
  - ID de imagen ECR: precargado, pero editable.

  Docker Hub image

  - Sistema operativo (SO) de la imagen: no se puede editar.
  - versión del sistema operativo: preseleccionado, pero editable.
  - ID de imagen de Docker: precargado, pero editable.

Configuración de instancias

- Fuente de la AMI (obligatoria): identifique una AMI personalizada para utilizarla como imagen base para la instancia de creación y prueba de su contenedor. Puede ser un ID de AMI o un parámetro del almacén de parámetros AWS Systems Manager (SSM) que contenga un ID de AMI.
  - ID de AMI: esta configuración no viene rellenada previamente con la entrada original. Introduzca el ID de AMI de la imagen base. Ejemplo: *ami-1234567890abcdef1*.
  - Parámetro SSM: introduzca el nombre o el ARN del parámetro SSM Parameter Store que contiene el ID de AMI de la imagen base. Ejemplo: */ib/test/param* o `arn:aws:ssm:`*us-east-1*`:`*111122223333*`:parameter`*/ib/test/param*.
- Almacenamiento (volúmenes)

  EBS volumen 1 (raíz AMI): precargado. No puede editar las selecciones de nombre de dispositivo, Instantánea o IOPS del volumen raíz. Sin embargo, puede cambiar todos los ajustes restantes, como el tamaño. También puede agregar nuevos volúmenes.

  > **Note**
  >
  > Si especificó una AMI base compartida con usted desde otra cuenta, las instantáneas de los volúmenes secundarios que se especifiquen también se deben compartir con su cuenta.

Directorio de trabajo

- Ruta del directorio de trabajo: precargada, pero editable.

Componentes

- Componentes: los componentes que ya están incluidos en la receta se muestran en la sección Componentes seleccionados al final de cada una de las listas de componentes (compilación y prueba). Puede eliminar o reordenar los componentes seleccionados para adaptarlos a sus necesidades.

  Los componentes de endurecimiento del CIS no siguen las reglas de ordenación de componentes estándar de las recetas de Generador de Imágenes. Los componentes de endurecimiento del CIS siempre se ejecutan en último lugar para garantizar que las pruebas de referencia se ejecuten con la imagen de salida.

> **Note**
>
> Las listas de componentes de compilación y prueba muestran los componentes disponibles según el tipo de propietario del componente. Para agregar un componente, elija Agregar componentes de compilación y seleccione el filtro de propiedad que corresponda. Por ejemplo, para añadir un componente de compilación asociado a un AWS Marketplace producto, seleccione`AWS Marketplace`. Esto abre un panel de selección en el lado derecho de la interfaz de la consola que muestra AWS Marketplace los componentes.
> Para el componente CIS, seleccione`Third party managed`.

Puede configurar las siguientes opciones para su componente seleccionado:

- Opciones de control de versiones: preseleccionadas, pero se pueden cambiar. Le recomendamos que elija la opción Usar la última versión del componente disponible para asegurarse de que las compilaciones de imágenes siempre incluyan la última versión del componente. Si necesita usar una versión de componente específica en su receta, puede elegir Especificar la versión del componente e introducir la versión en el cuadro versión del componente que aparece.
- Parámetros de entrada: muestra los parámetros de entrada que acepta el componente. El valor está precargado con el valor de la versión anterior de la receta. Si utiliza este componente por primera vez en esta receta y se definió un valor predeterminado para el parámetro de entrada, el valor predeterminado aparece en el cuadro Valor con el texto atenuado. Si no se especifica ningún otro valor, Generador de imágenes utiliza el valor predeterminado.

  Si se requiere un parámetro de entrada, pero no tiene un valor predeterminado definido en el componente, debe proporcionar un valor. Generador de imágenes no creará la versión de la receta si falta algún parámetro obligatorio y no tiene definido un valor predeterminado.

  > **Important**
  >
  > Los parámetros del componente son valores de texto sin formato y se registran en AWS CloudTrail. Le recomendamos que utilice AWS Secrets Manager el almacén de AWS Systems Manager parámetros para almacenar sus secretos. Para obtener más información sobre Secrets Manager, consulte ¿Qué es Secrets Manager? en la Guía del usuario de AWS Secrets Manager . Para obtener más información acerca del almacén

de parámetros de AWS Systems Manager , consulte Almacén de parámetros de AWS Systems Manager en la Guía del usuario de AWS Systems Manager .

Para ampliar la configuración de las opciones de control de versiones o los parámetros de entrada, puede seleccionar la flecha junto al nombre de la configuración. Para ampliar todos los ajustes de todos los componentes seleccionados, puede activar y desactivar el interruptor Expandir todo.

Plantilla de Dockerfile

- Plantilla de Dockerfile: precargada, pero editable. Puede especificar cualquiera de las siguientes variables contextuales que Generador de imágenes reemplaza por información de compilación en tiempo de ejecución.

  parentImage (obligatorio)

  En el momento de la compilación, esta variable se convierte en la imagen base de la receta.

  Ejemplo:

  ```
  FROM
  {{{ imagebuilder:parentImage }}}
  ```

  entornos (obligatorio si se especifican los componentes)

  Esta variable se convertirá en un script que ejecuta componentes.

  Ejemplo:

  ```
  {{{ imagebuilder:environments }}}
  ```

  componentes (opcional)

  Generador de imágenes resuelve los scripts de componentes de compilación y prueba para los componentes que incluye la receta del contenedor. Esta variable se puede colocar en cualquier parte del Dockerfile, después de la variable de entorno.

  Ejemplo:

```
{{{ imagebuilder:components }}}
```

Repositorio de destino

- Nombre del repositorio de destino: el repositorio de Amazon ECR donde se almacena la imagen de salida si no hay otro repositorio especificado en la configuración de distribución de la canalización para la región donde se ejecuta la canalización (región 1).

Para crear una nueva versión de una receta de contenedor:

1. En la parte superior de la página de detalles de la receta de contenedor, seleccione Crear nueva versión. Accederá a la página Crear receta para las recetas de contenedor.
2. Para crear la nueva versión, realice los cambios y, a continuación, elija Crear receta.

Para obtener más información sobre cómo crear una receta de contenedor cuando crea una canalización de imágenes, consulte Paso 2: elegir receta en la sección Introducción de esta guía.

Crear una receta de contenedor con la AWS CLI

Para crear una receta de contenedor de Image Builder con el `imagebuilder create-container-recipe` comando de AWS CLI, siga estos pasos:

Requisitos previos

Antes de ejecutar los comandos de Image Builder de esta sección para crear una receta de contenedor con el AWS CLI, debe crear los componentes que utilizará la receta. El ejemplo de receta de contenedor en el siguiente paso hace referencia a los componentes de ejemplo que se crean en la sección Cree un componente personalizado a partir del AWS CLI de esta guía.

Tras crear los componentes, o si utiliza componentes existentes, anote los ARNs que desee incluir en la receta.

1. Creación de un archivo JSON de entrada de la CLI

   Puede proporcionar todas las entradas para el comando create-container-recipe con parámetros de comando en línea. Sin embargo, el comando resultante puede ser bastante largo.

Para simplificar el comando, puede proporcionar un archivo JSON que contenga todas las configuraciones de la receta de contenedor.

> **Note**
>
> La convención de nomenclatura de los valores de datos del archivo JSON sigue el patrón que se especifica para los parámetros de solicitud de operación de la API Image Builder. Para revisar los parámetros de la solicitud de operación de la API, consulte el CreateContainerRecipecomando en la Referencia de la API de EC2 Image Builder. Para proporcionar los valores de los datos como parámetros de la línea de comandos, consulte los nombres de los parámetros especificados en la referencia de comandos de la AWS CLI .

Este es un resumen de los parámetros de este ejemplo:

- Componentes (matriz de objetos, obligatorio): contiene un conjunto de objetos `ComponentConfiguration`. Debe especificarse al menos un componente de compilación:

  > **Note**
  >
  > Generador de Imágenes instala los componentes en el orden en que se especificaron en la receta. Sin embargo, los componentes de refuerzo del CIS siempre se ejecutan en último lugar para garantizar que las pruebas de referencia se ejecutan con su imagen de salida.

  - componentARN (cadena, obligatorio): el ARN del componente.

    > **Tip**
    >
    > Si desea utilizar el ejemplo para crear su propia receta de recipiente, sustituya el ejemplo ARNs ARNs por el de los componentes que está utilizando para la receta. Estos incluyen la Región de AWS, el nombre y el número de versión de cada uno.

  - Parámetros (matriz de objetos): contiene una matriz de objetos `ComponentParameter`. Si se requiere un parámetro de entrada, pero no tiene un valor predeterminado definido en el

componente, debe proporcionar un valor. Generador de imágenes no creará la versión de la receta si falta algún parámetro obligatorio y no tiene definido un valor predeterminado.

> **Important**
>
> Los parámetros del componente son valores de texto sin formato y se registran en AWS CloudTrail. Le recomendamos que utilice AWS Secrets Manager nuestro almacén de AWS Systems Manager parámetros para almacenar sus secretos. Para obtener más información sobre Secrets Manager, consulte ¿Qué es Secrets Manager? en la Guía del usuario de AWS Secrets Manager . Para obtener más información acerca del almacén de parámetros de AWS Systems Manager , consulte Almacén de parámetros de AWS Systems Manager en la Guía del usuario de AWS Systems Manager .

    - Nombre (cadena, obligatorio): el nombre del parámetro del componente que se va a establecer.
    - valor (matriz de cadenas, obligatorio): contiene una matriz de cadenas para establecer el valor del parámetro de componente nombrado. Si hay un valor por defecto definido para el componente y no se proporciona ningún otro valor, TOE de AWS utiliza el valor por defecto.
- containerType (cadena, obligatorio): el tipo de contenedor que se va a crear. Los valores válidos incluyen `DOCKER`.
- dockerfileTemplateData(cadena): la plantilla de Dockerfile que se utiliza para crear la imagen, expresada como un blob de datos en línea.
- Nombre (cadena, obligatorio): nombre de la receta de contenedor.
- Descripción (cadena): descripción de la receta de contenedor.
- parentImage (cadena, obligatorio): la imagen del contenedor de Docker que se utilizará en la receta del contenedor como base para la imagen personalizada.
    - Imágenes públicas alojadas en DockerHub
    - Imágenes de contenedores existentes en Amazon ECR
    - Imágenes de contenedores gestionadas por Amazon
- platformOverride (cadena): especifica la plataforma del sistema operativo cuando se utiliza una imagen base personalizada.

- semanticVersion (cadena, obligatorio): la versión semántica de la receta de contenedor especificada en el siguiente formato, con valores numéricos en cada posición para indicar una versión específica: <major>.<minor>.<patch>. Un ejemplo sería `1.0.0`. Para obtener más información sobre el control de versiones semántico para los recursos de Generador de Imágenes, consulte Control de versiones semántico en Generador de imágenes.
- Etiquetas (mapa de cadenas): etiquetas que se adjuntan a la receta de contenedor.
- instanceConfiguration (objeto): un grupo de opciones que se pueden usar para configurar una instancia para compilar y probar las imágenes de contenedor.
  - image (string): la imagen base de una instancia de compilación y prueba de un contenedor. Puede contener un ID de AMI o puede especificar un parámetro del almacén de parámetros AWS Systems Manager (SSM), con el prefijo y seguido del nombre del parámetro o el ARN. `ssm:` Si utiliza un parámetro SSM, el valor del parámetro debe contener un ID de AMI. Si no especifica una imagen base, Image Builder utiliza la AMI optimizada para Amazon ECS adecuada como imagen base.
  - blockDeviceMappings(matriz de objetos): define los dispositivos de bloques que se van a adjuntar para crear una instancia a partir de la AMI de Image Builder especificada en el image parámetro.
    - deviceName (cadena): el dispositivo al que se aplican estas asignaciones.
    - ebs (objeto): se utiliza para administrar la configuración específica de Amazon EBS para esta asignación.
      - deleteOnTermination(Booleano): se utiliza para configurar la eliminación al finalizar el dispositivo asociado.
      - cifrado (Booleano): se utiliza para configurar el cifrado del dispositivo.
      - volumeSize (entero): se utiliza para anular el tamaño de volumen del dispositivo.
      - volumeType (cadena): se utiliza para anular el tipo de volumen del dispositivo.
- targetRepository (objeto, obligatorio): el repositorio de destino de la imagen de contenedor si no hay ningún otro repositorio especificado en la configuración de distribución de la canalización para la región donde se ejecuta la canalización (región 1).
  - repositoryName (cadena, obligatorio): el nombre del repositorio de contenedor donde se almacena la imagen del contenedor de salida. Este nombre está precedido por la ubicación del repositorio.
  - service (cadena, obligatorio): especifica el servicio en el que se registró esta imagen.

- workingDirectory (cadena): directorio de trabajo que se utilizará durante los flujos de trabajo de compilación y prueba.

```
{
 "components": [
   {
   "componentArn": "arn:aws:imagebuilder:us-west-2:111122223333:component/
helloworldal2/x.x.x"
   }
 ],
 "containerType": "DOCKER",
 "description": "My Linux Docker container image",
 "dockerfileTemplateData": "FROM
 {{{ imagebuilder:parentImage }}}\n{{{ imagebuilder:environments }}}\n{{{ imagebuilder:comp
 "name": "amazonlinux-container-recipe",
 "parentImage": "amazonlinux:latest",
 "platformOverride": "Linux",
 "semanticVersion": "1.0.2",
 "tags": {
   "sometag" : "Tag detail"
 },
 "instanceConfiguration": {
   "image": "ami-1234567890abcdef1",
   "blockDeviceMappings": [
   {
   "deviceName": "/dev/xvda",
   "ebs": {
    "deleteOnTermination": true,
    "encrypted": false,
    "volumeSize": 8,
    "volumeType": "gp2"
    }
    }
   ]
 },
 "targetRepository": {
   "repositoryName": "myrepo",
   "service": "ECR"
 },
 "workingDirectory": "/tmp"
}
```

2. Creación de la receta

   Utilice el siguiente comando para crear la receta. Introduzca el nombre del archivo JSON que creó en el paso anterior en el parámetro `--cli-input-json`:

   ```
   aws imagebuilder create-container-recipe --cli-input-json file://create-container-recipe.json
   ```

   > **Note**
   >
   > - Debe incluir la notación `file://` al principio de la ruta del archivo JSON.
   > - La ruta del archivo JSON debe seguir la convención apropiada para el sistema operativo base donde se está ejecutando el comando. Por ejemplo, Windows utiliza la barra diagonal inversa (\) para referirse a la ruta de directorios, mientras que Linux y macOS usan la barra diagonal (/).

Eliminar recursos

Para evitar cargos inesperados, asegúrese de limpiar los recursos y canalizaciones que haya creado a partir de los ejemplos de esta guía. Para obtener más información sobre cómo eliminar recursos en Generador de Imágenes, consulte Eliminación de recursos de Generador de imágenes obsoletos o no utilizados.

Administre la configuración de la infraestructura de Generador de imágenes

Puede utilizar las configuraciones de infraestructura para especificar la EC2 infraestructura de Amazon que Image Builder utiliza para crear y probar su EC2 imagen de Image Builder. La configuración de la infraestructura incluye:

- Tipos de instancias para la infraestructura de construcción y prueba. Te recomendamos que especifiques más de un tipo de instancia, ya que esto permite a Generador de Imágenes lanzar una instancia desde un grupo con capacidad suficiente. Esto puede reducir los errores de compilación transitorios.

En el caso de las imágenes de Mac, puede que desee elegir tipos de instancia que admitan de forma nativa el sistema operativo macOS. Para obtener más información, consulta las instancias de Amazon EC2 Mac en la Guía del EC2 usuario de Amazon.

- Configuraciones de ubicación de instancias donde puede especificar el host, el grupo de ubicación de hosts o la zona de disponibilidad a los que deben asignarse las instancias que se inicien desde su imagen.
- Un perfil de instancia que proporciona a las instancias de compilación y prueba los permisos necesarios para realizar actividades de personalización. Por ejemplo, si tiene un componente que recupera recursos de Amazon S3, el perfil de instancia requiere permisos para acceder a esos archivos. El perfil de instancia también requiere un conjunto mínimo de permisos para que EC2 Image Builder se comunique correctamente con la instancia. Para obtener más información, consulte Preparación para crear imágenes personalizadas con Generador de imágenes.
- La VPC, la subred y los grupos de seguridad de las instancias de compilación y prueba de su canalización.
- La ubicación de Amazon S3 en la que Generador de Imágenes almacena los registros de las aplicaciones de su compilación y pruebas. Si configura el registro, el perfil de instancia especificado en la configuración de la infraestructura debe tener permisos `s3:PutObject` para el bucket de destino (`arn:aws:s3:::BucketName/*`).
- Un par de EC2 claves de Amazon que te permite iniciar sesión en tu instancia para solucionar problemas en caso de que la compilación falle y lo hayas configurado`terminateInstanceOnFailure`. `false`
- Un tema de SNS en el que Generador de Imágenes envía notificaciones de eventos. Para obtener más información sobre la integración de Generador de Imágenes con Amazon SNS, consulte Integración de Amazon SNS en Generador de Imágenes.

> **Note**
>
> Si su tema de SNS está cifrado, la clave que cifra este tema debe residir en la cuenta en la que se ejecuta el servicio de Generador de Imágenes. Generador de Imágenes no puede enviar notificaciones a temas de SNS que estén cifrados con claves de otras cuentas.

Puede crear y administrar configuraciones de infraestructura mediante la consola de administración Generador de Imágenes API o con los comandos imagebuilder en la AWS CLI.

Contenido

- Enumeración y visualización de los detalles de configuración de la infraestructura
- Crear una configuración de infraestructura
- Actualización de una configuración de infraestructura
- Image Builder y puntos finales AWS PrivateLink de VPC de interfaz

> **Tip**
>
> Cuando tenga muchos recursos del mismo tipo, el etiquetado puede ayudarle a identificar un recurso específico en función de las etiquetas que le haya asignado. Para obtener más información sobre cómo etiquetar los recursos mediante los comandos de Image Builder de AWS CLI, consulte la Etiquetar recursos sección de esta guía.

Enumeración y visualización de los detalles de configuración de la infraestructura

En esta sección se describen las distintas formas en las que puede encontrar información y ver los detalles de las configuraciones de infraestructura de EC2 Image Builder.

Detalles de la configuraciones de infraestructura

- Enumere las configuraciones de infraestructura del AWS CLI
- Obtenga los detalles de configuración de la infraestructura en el AWS CLI

Enumere las configuraciones de infraestructura del AWS CLI

En el siguiente ejemplo se muestra cómo enumerar todas las configuraciones de la infraestructura mediante el comando list-infrastructure-configurations en la AWS CLI.

```
aws imagebuilder list-infrastructure-configurations
```

Obtenga los detalles de configuración de la infraestructura en el AWS CLI

El siguiente ejemplo muestra cómo usar el get-infrastructure-configuration comando AWS CLI para obtener los detalles de una configuración de infraestructura especificando su nombre de recurso de Amazon (ARN).

```
aws imagebuilder get-infrastructure-configuration --infrastructure-configuration-arn
 arn:aws:imagebuilder:us-west-2:123456789012:infrastructure-configuration/my-example-
infrastructure-configuration
```

Crear una configuración de infraestructura

En esta sección se describe cómo puede utilizar la consola o los imagebuilder comandos de Image Builder AWS CLI para crear una configuración de infraestructura,

Console

Para crear un recurso de configuración de infraestructura desde la consola de Generador de Imágenes, siga estos pasos:

1. Abra la consola EC2 de Image Builder en https://console.aws.amazon.com/imagebuilder/.
2. En el panel de navegación, elija Configuración de la infraestructura..
3. Elija Crear configuración de infraestructura.
4. En la sección General, introduzca las siguientes casillas obligatorias:
   - Introduzca el nombre del recurso de configuración de la infraestructura.
   - Seleccione un rol de IAM que desee asociar al perfil de instancia para los permisos de los componentes de las instancias de creación y prueba. Image Builder utiliza estos permisos para descargar y ejecutar los componentes CloudWatch, cargar registros y realizar cualquier acción adicional que especifiquen los componentes de la receta.
5. En el panel de infraestructura de AWS, puede configurar los ajustes de infraestructura restantes que estén disponibles. Introduzca la siguiente información necesaria:
   - Tipo de instancia: puede especificar uno o más tipos de instancia para utilizarlos en esta compilación. El servicio seleccionará uno de estos tipos de instancias en función de la disponibilidad.

     > **Note**
     >
     > Las instancias de Mac se ejecutan en tipos de instancia `.metal` en un host dedicado. El tipo de instancia debe coincidir con uno de los tipos definidos para el host en el que se ejecuta. Para obtener más información sobre las instancias de Mac y una lista de los tipos de instancias compatibles de forma nativa con el

sistema operativo macOS, consulta las instancias de Amazon EC2 Mac en la Guía del EC2 usuario de Amazon.

- Tema de SNS (opcional): seleccione un tema de SNS para recibir notificaciones y alertas de EC2 Image Builder.

Si no proporcionas valores para las siguientes configuraciones, usarán los valores predeterminados específicos del servicio, cuando corresponda.

- VPC, subred y grupos de seguridad: Generador de Imágenes usa la VPC y la subred predeterminadas. Para obtener más información sobre cómo configurar el punto de conexión de interfaz, consulte Image Builder y puntos finales AWS PrivateLink de VPC de interfaz.
- En la sección Configuración de solución de problemas, puede configurar los siguientes valores:
  - De forma predeterminada, está activada la casilla de verificación Finalizar instancia en caso de falla. Sin embargo, cuando se produce un error en una compilación, puedes iniciar sesión en la EC2 instancia para solucionar el problema. Si desea que la instancia siga ejecutándose después de un error de compilación, desactive la casilla.
  - Par de claves: si la EC2 instancia sigue ejecutándose después de un error de compilación, puedes crear un par de claves o usar un par de claves existente para iniciar sesión en la instancia y solucionar el problema.
  - Registros: puede especificar un bucket de S3 en el que Generador de Imágenes pueda escribir los registros de las aplicaciones para solucionar problemas de compilación y pruebas. Si no especifica un bucket de S3, Generador de Imágenes escribirá los registros de la aplicación en la instancia.
- En la sección Configuración de metadatos de instancias, puede configurar los siguientes valores para aplicarlos a las EC2 instancias que Image Builder utiliza para crear y probar la imagen:
  - Seleccione la versión de metadatos para determinar si EC2 requiere un encabezado de token firmado para, por ejemplo, las solicitudes de recuperación de metadatos.
    - V1 y V2 (token opcional): valor predeterminado si no selecciona nada.
    - V2 (token obligatorio)

> **Note**
>
> Le recomendamos que configure todas las EC2 instancias que Image Builder lance desde una compilación de canalización para IMDSv2 que las solicitudes de recuperación de metadatos de las instancias requieran un encabezado de token firmado.

- Límite de saltos de respuesta del token de metadatos: el número saltos de red que puede recorrer el token de metadatos. Saltos mínimos: 1, saltos máximos: 64, con un salto como valor predeterminado.

- En la sección Configuración de ubicación de instancias, puede configurar los siguientes valores para aplicarlos a las EC2 instancias que Image Builder utiliza para crear y probar la imagen:
  - Puede seleccionar la zona de disponibilidad en la que Generador de imágenes lanza las instancias durante la creación de la imagen.
  - Si lo desea, seleccione Tenencia para los servidores que ejecutan las instancias que lance. De forma predeterminada, EC2 las instancias se ejecutan en hardware de arrendamiento compartido. Esto significa que varias Cuentas de AWS pueden compartir el mismo hardware físico. Una instancia con una tenencia de ejecuciones `dedicated` en hardware de un solo propietario. Una instancia con una tenencia de `host` se ejecuta en un host dedicado.

    Las instancias de Mac requieren que se cree un host dedicado como requisito previo antes de crear una imagen personalizada. Seleccione `host` para su imagen de macOS. A continuación, puede seleccionar un host o un grupo de recursos de hosts de destino para lanzar instancias, pero no es obligatorio si su host dedicado tiene habilitada la ubicación automática. Para obtener más información, consulta Colocación automática en la Guía del EC2 usuario de Amazon.
    - ID de host de tenencia: el ID del host dedicado en el que se ejecutan las instancias.
    - Grupo de recursos de host de tenencia: nombre de recurso de Amazon (ARN) del grupo con recursos de host en el que se iniciarán las instancias.

6. En la sección Etiquetas de infraestructura (opcional), puede asignar etiquetas de metadatos a la EC2 instancia de Amazon que Image Builder lanza durante el proceso de creación. Las etiquetas se introducen como pares clave-valor.

7. En la sección Etiquetas (opcional), puede asignar etiquetas de metadatos al recurso de configuración de infraestructura que Generador de imágenes crea como salida. Las etiquetas se introducen como pares clave-valor.

AWS CLI

El siguiente procedimiento muestra cómo configurar la infraestructura de la imagen con el comando create-infrastructure-configuration de Generador de iImágenes en la AWS CLI. El comando del paso 2 incluye el archivo que creó en el paso 1. Para estos ejemplos, el archivo del paso 1 se denomina `create-infrastructure-configuration.json`.

1. Creación de un archivo JSON de entrada de la CLI

   Los siguientes ejemplos muestran variaciones del archivo JSON que puede crear para la configuración de la infraestructura. Utilice una herramienta de edición de archivos para crear un archivo JSON de su propiedad.

   Ejemplo 1: configuración para retener una instancia de una compilación fallida

   En este ejemplo se especifican dos tipos de instancias, `m5.large` y `m5.xlarge`. Te recomendamos que especifiques más de un tipo de instancia, ya que esto permite a Generador de Imágenes lanzar una instancia desde un grupo con capacidad suficiente. Esto puede reducir los errores de compilación transitorios.

   `instanceProfileName` especifica el perfil de instancia que proporciona a la instancia los permisos que el perfil necesita para realizar actividades de personalización. Por ejemplo, si tiene un componente que recupera recursos de Amazon S3, el perfil de instancia requiere permisos para acceder a esos archivos. El perfil de instancia también requiere un conjunto mínimo de permisos para que EC2 Image Builder se comunique correctamente con la instancia. Para obtener más información, consulte Preparación para crear imágenes personalizadas con Generador de imágenes.

   ```
   {
       "name": "ExampleInfraConfigDontTerminate",
       "description": "An example that will retain instances of failed builds",
       "instanceTypes": [
           "m5.large", "m5.xlarge"
       ],
       "instanceProfileName": "myIAMInstanceProfileName",
       "securityGroupIds": [
   ```

```
        "sg-12345678"
    ],
    "subnetId": "sub-12345678",
    "logging": {
        "s3Logs": {
            "s3BucketName": "my-logging-bucket",
            "s3KeyPrefix": "my-path"
        }
    },
    "keyPair": "myKeyPairName",
    "terminateInstanceOnFailure": false,
    "snsTopicArn": "arn:aws:sns:us-west-2:123456789012:MyTopic"
}
```

Ejemplo 2: configuración de macOS con ubicación automática

En este ejemplo, se especifican los tipos de instancia y la ubicación de una instancia de Mac en la que el host dedicado tiene habilitada la ubicación automática.

```
{
   "name": "macOSInfraConfigAutoPlacement",
   "description": "An example infrastructure configuration for macOS.",
   "instanceProfileName": "EC2InstanceProfileForImageBuilder",
   "instanceTypes": ["mac1.metal, mac2.metal"],
   "terminateInstanceOnFailure": false,
   "placement": {
      "tenancy": "host"
   }
}
```

Ejemplo 3: configuración de macOS con el ID de host especificado

En este ejemplo, se especifica el tipo de instancia y la ubicación de una instancia de Mac que se dirige a un host dedicado específico.

```
{
   "name": "macOSInfraConfigHostPlacement",
   "description": "An example infrastructure configuration for macOS.",
   "instanceProfileName": "EC2InstanceProfileForImageBuilder",
   "instanceTypes": ["mac2-m1ultra.metal"],
   "terminateInstanceOnFailure": false,
   "placement": {
```

```
      "tenancy": "host",
      "hostId" : "h-1234567890abcdef0"
   }
}
```

2. Utilice el archivo creado como entrada cuando ejecute el siguiente comando.

```
aws imagebuilder create-infrastructure-configuration --cli-input-json
 file://create-infrastructure-configuration.json
```

Actualización de una configuración de infraestructura

En esta sección se explica cómo puede utilizar la consola o los imagebuilder comandos de Image Builder AWS CLI para actualizar un recurso de configuración de infraestructura. Para realizar un seguimiento de sus recursos, puede aplicar etiquetas de la siguiente manera. Las etiquetas se introducen como pares clave-valor.

- Las etiquetas de recursos asignan etiquetas de metadatos a la EC2 instancia de Amazon que Image Builder lanza durante el proceso de compilación.
- Las etiquetas asignan etiquetas de metadatos al recurso de configuración de infraestructura que Generador de imágenes crea como salida.

Console

Puede editar los siguientes detalles de configuración de la infraestructura desde la consola de Generador de Imágenes:

- La descripción de la configuración de la infraestructura.
- El rol de IAM que asociará con el perfil de instancia.
- AWS infraestructura, incluido el tipo de instancia y un tema de SNS para las notificaciones.
- VPC, la subred y los grupos de seguridad.
- La configuración de solución de problemas, incluida la finalización de la instancia en caso de falla, el par de claves para la conexión y una ubicación de bucket de S3 opcional para los registros de las instancias.

Para actualizar un recurso de configuración de infraestructura desde la consola de Generador de Imágenes, siga estos pasos:

Elija una configuración de infraestructura de Generador de Imágenes existente.

1. Abra la consola EC2 de Image Builder en https://console.aws.amazon.com/imagebuilder/.
2. Para ver una lista de los recursos de configuración de infraestructura de su cuenta, seleccione Configuración de infraestructura en el panel de navegación.
3. Para ver los detalles o editar una configuración de infraestructura, elija el enlace nombre de la configuración. Esto abre la vista detallada de la configuración de la infraestructura.

> **Note**
>
> También puede seleccionar la casilla junto al nombre de la configuración, y, a continuación, elija Ver detalles.

4. En la esquina superior derecha del panel de detalles de la infraestructura, seleccione Editar.
5. Cuando esté listo para guardar las actualizaciones que ha realizado en la configuración de la infraestructura, seleccione Guardar cambios.

AWS CLI

El siguiente ejemplo muestra cómo actualizar la infraestructura de la imagen con el comando update-infrastructure-configuration de Generador de imágenes en la AWS CLI.

1. Creación de un archivo JSON de entrada de la CLI

   En este ejemplo de configuración de infraestructura se utilizan los mismos parámetros que en el ejemplo de creación, con la salvedad de que hemos actualizado el ajuste `terminateInstanceOnFailure` a `false`. Después de ejecutar el comando update-infrastructure-configuration, las canalizaciones que utilizan esta configuración de infraestructura finalizan las instancias de compilación y prueba cuando la compilación falla.

   Utilice una herramienta de edición de archivos para crear un archivo JSON con las claves que se muestran en el siguiente ejemplo, además de valores que sean válidos para su entorno. En este ejemplo, se utiliza un archivo con el nombre `update-infrastructure-configuration.json`:

```
{
```

```
"infrastructureConfigurationArn": "arn:aws:imagebuilder:us-
west-2:123456789012:infrastructure-configuration/my-example-infrastructure-
configuration",
"description": "An example that will terminate instances of failed builds",
"instanceTypes": [
    "m5.large", "m5.2xlarge"
],
"instanceProfileName": "myIAMInstanceProfileName",
"securityGroupIds": [
    "sg-12345678"
],
"subnetId": "sub-12345678",
"logging": {
    "s3Logs": {
        "s3BucketName": "my-logging-bucket",
        "s3KeyPrefix": "my-path"
    }
},
"terminateInstanceOnFailure": true,
"snsTopicArn": "arn:aws:sns:us-west-2:123456789012:MyTopic"
}
```

2. Utilice el archivo creado como entrada cuando ejecute el siguiente comando.

```
aws imagebuilder update-infrastructure-configuration --cli-input-json
 file://update-infrastructure-configuration.json
```

Image Builder y puntos finales AWS PrivateLink de VPC de interfaz

Puede establecer una conexión privada entre su VPC e EC2 Image Builder mediante la creación de un punto final de interfaz de VPC. Los puntos finales de la interfaz funcionan con una tecnología que le permite acceder de forma privada a Image Builder APIs sin una puerta de enlace a Internet, un dispositivo NAT, una conexión VPN o una Direct Connect conexión. AWS PrivateLink Las instancias de su VPC no necesitan direcciones IP públicas para comunicarse con Image Builder. APIs El tráfico entre su VPC e Generador de Imágenes no sale de la red de Amazon.

Cada punto de conexión de la interfaz está representado por una o más interfaces de red elásticas en las subredes. Al crear una imagen nueva, puede especificar el ID de subred de la VPC en la configuración de la infraestructura.

> **Note**
>
> Cada servicio al que se accede desde una VPC tiene su propio punto de conexión de interfaz, con su propia política de punto de conexión. Image Builder descarga la aplicación de administrador de TOE de AWS componentes y accede a los recursos gestionados desde los buckets de S3 para crear imágenes personalizadas. Para conceder el acceso a esos buckets, debe actualizar la política de puntos de conexión de S3 para permitirlo. Para obtener más información, consulte Políticas personalizadas para el acceso al bucket de S3.

Para obtener más información sobre puntos de conexión de VPC, consulte Puntos de conexión de VPC de interfaz (AWS PrivateLink) en la Guía del usuario de Amazon VPC.

Consideraciones sobre los puntos de conexión de VPC de Generador de Imágenes

Antes de configurar un punto de conexión de VPC de interfaz para Generador de Imágenes, asegúrese de revisar las propiedades y limitaciones de los puntos de conexión de interfaz en la Guía del usuario de Amazon VPC.

Generador de Imágenes admite realizar llamadas a todas sus acciones de la API desde su VPC.

Creación de un punto de conexión de VPC de interfaz para Generador de Imágenes

Para crear un punto final de VPC para el servicio Image Builder, puede utilizar la consola de Amazon VPC o el ().AWS Command Line InterfaceAWS CLI Para obtener más información, consulte Creación de un punto de conexión de interfaz en la Guía del usuario de Amazon VPC.

Cree un punto de conexión de VPC para Generador de Imágenes mediante el siguiente nombre de servicio:

- com.amazonaws. *region*.creador de imágenes

Si habilita DNS privado para el punto de conexión, puede realizar solicitudes API a Generador de Imágenes usando su nombre de DNS predeterminado para la región, por ejemplo, `imagebuilder.us-east-1.amazonaws.com`. Para buscar el punto de conexión que se aplica a la región de destino, consulte los puntos de enlace y las cuotas de EC2 Image Builder en. Referencia general de Amazon Web Services

Para más información, consulte Acceso a un servicio a través de un punto de conexión de interfaz en la Guía del usuario de Amazon VPC.

Creación de una política de puntos de conexión de VPC para Generador de Imágenes

Puede asociar una política de puntos de conexión con su punto de conexión de VPC que controla el acceso a Generador de Imágenes. La política especifica la siguiente información:

- La entidad principal que puede realizar acciones.
- Las acciones que se pueden realizar.
- Los recursos en los que se pueden llevar a cabo las acciones.

Si utiliza componentes administrados por Amazon en su receta, el punto de conexión de VPC para Generador de Imágenes debe permitir el acceso a la siguiente biblioteca de componentes propiedad del servicio:

`arn:aws:imagebuilder:region:aws:component/*`

> **Important**
>
> Cuando se aplica una política no predeterminada a un punto final de la interfaz de VPC para Image EC2 Builder, es posible que algunas solicitudes de API con errores, como las que no lleguen`RequestLimitExceeded`, no se registren en Amazon AWS CloudTrail. CloudWatch

Para más información, consulte Control del acceso a los servicios con puntos de conexión de VPC en la Guía del usuario de Amazon VPC.

Políticas personalizadas para el acceso al bucket de S3

Generador de Imágenes utiliza un bucket de S3 disponible públicamente para almacenar y acceder a los recursos administrados, como los componentes. También descarga la aplicación de administración de TOE de AWS componentes desde un bucket de S3 independiente. Si utiliza un punto de conexión de VPC para Amazon S3 en su entorno, deberá asegurarse de que su política de punto de conexión de VPC de S3 permita a Generador de Imágenes acceder a los siguientes buckets de S3. Los nombres de los buckets son únicos por AWS región (*region*) y por entorno de aplicación (*environment*). Image Builder y TOE de AWS son compatibles con los siguientes entornos de aplicaciones: `prod`preprod, y`beta`.

- El depósito TOE de AWS del administrador de componentes:

```
s3://ec2imagebuilder-toe-region-environment
```

Ejemplo: s3://ec2 imagebuilder-toe-us-west -2-prod/*

- El bucket de recursos administrados por Generador de Imágenes:

```
s3://ec2imagebuilder-managed-resources-region-environment/components
```

Ejemplo: s3://ec2 -west-2-prod/components/* imagebuilder-managed-resources-us

Ejemplos de políticas de puntos de conexión de VPC

En esta sección se incluyen ejemplos de políticas de puntos de conexión de VPC personalizadas.

Política general de puntos de conexión de VPC para acciones de Generador de Imágenes

El siguiente ejemplo de política de punto de conexión para Generador de Imágenes deniega el permiso para eliminar imágenes y componentes de Generador de Imágenes. La política de ejemplo también concede permiso para realizar todas las demás acciones EC2 de Image Builder.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Action": "imagebuilder:*",
            "Effect": "Allow",
            "Resource": "*"
        },
        {
            "Action": "imagebuilder:DeleteImage",
            "Effect": "Deny",
            "Resource": "*"
        },
        {
            "Action": "imagebuilder:DeleteComponent",
            "Effect": "Deny",
            "Resource": "*"
        }
```

```
    ]
}
```

Restricción del acceso por organización, permiso a acceder a los componentes administrados

El siguiente ejemplo de política de punto de conexión ilustra cómo restringir el acceso a las identidades y los recursos que pertenecen a su organización y proporcionar acceso a los componentes de Generador de imágenes administrados por Amazon. Sustituya *region* y por *resource-org-id* los valores de su organización. *principal-org-id*

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AllowRequestsByOrgsIdentitiesToOrgsResources",
            "Effect": "Allow",
            "Principal": {
                "AWS": "*"
            },
            "Action": "*",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:PrincipalOrgID": "principal-org-id",
                    "aws:ResourceOrgID": "resource-org-id"
                }
            }
        },
        {
            "Sid": "AllowAccessToEC2ImageBuilderComponents",
            "Effect": "Allow",
            "Principal": {
                "AWS": "*"
            },
            "Action": [
                "imagebuilder:GetComponent"
            ],
            "Resource": [
                "arn:aws:imagebuilder:us-east-1:aws:component/*"
```

```
            ]
        }
    ]
}
```

Política de punto de conexión de VPC para acceso al bucket de Amazon S3

El siguiente ejemplo de política de punto de conexión de S3 muestra cómo proporcionar acceso a los buckets de S3 que Generador de Imágenes utiliza para crear imágenes personalizadas. Sustituya *region* y *environment* con los valores de su organización. Añada cualquier otro permiso necesario a la política en función de los requisitos de su aplicación.

> **Note**
>
> En el caso de las imágenes de Linux, si no especifica los datos de usuario en la receta de la imagen, Generador de imágenes añade un script para descargar e instalar el agente de Systems Manager en las instancias de compilación y prueba de la imagen. Para descargar el agente, Generador de imágenes accede al bucket de S3 de su región de compilación. Para garantizar que Generador de imágenes pueda iniciar las instancias de compilación y prueba, añada el siguiente recurso adicional a su política de puntos de conexión de S3: `"arn:aws:s3:::amazon-ssm-region/*"`

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Sid": "AllowImageBuilderAccessToAppAndComponentBuckets",
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
      },
      "Action": [
        "s3:GetObject"
      ],
      "Resource": [
        "arn:aws:s3:::ec2imagebuilder-toe-region-environment/*",
```

```
        "arn:aws:s3:::ec2imagebuilder-managed-resources-region-environment/
components/*"
      ]
    }
  ]
}
```

Administrar los ajustes de la distribución de Generador de imágenes

Antes de configurar los ajustes de distribución de las imágenes de salida, le recomendamos que compruebe la disponibilidad de cualquier infraestructura subyacente u otros requisitos para las instancias que se lancen desde la imagen de salida en las regiones de destino de la distribución. Por ejemplo, no todas las regiones admiten hosts dedicados de EC2 Mac, que son necesarios para lanzar instancias desde una imagen de macOS. Para obtener más información sobre los tipos de instancias y los precios de los hosts dedicados, consulta los precios de los hosts EC2 dedicados de Amazon.

Tras crear los ajustes de distribución con Generador de Imágenes, puede administrarla mediante la consola Generador de Imágenes, la API Generador de Imágenes o los comandos imagebuilder de la AWS CLI. Con los ajustes de distribución, puede realizar las siguientes acciones:

Distribución de AMI

- Especifique el nombre y la descripción de su AMI de salida.
- Autorizar a otras Cuentas de AWS organizaciones y OUs lanzar la AMI desde la cuenta del propietario. A la cuenta del propietario se le facturan los cargos asociados a la AMI.

> **Note**
>
> Para hacer una AMI pública, configure las cuentas autorizadas de permiso de lanzamiento en `all`. Para obtener información y ejemplos, consulta ModifyImageAttribute la referencia de la EC2 API de Amazon.

- Cree una copia de la AMI de salida para cada una de las cuentas y organizaciones de destino especificadas y OUs en la región de destino. Las cuentas y OUs organizaciones objetivo son propietarios de sus copias de la AMI y se les facturan los cargos asociados. Para obtener más

información sobre la distribución de la AMI a AWS Organizations y OUs, consulte Compartir una AMI con organizaciones o OUs.

- Copie la AMI a la cuenta del propietario en otra Regiones de AWS.
- Exporte discos de imágenes de máquinas virtuales a Amazon Simple Storage Service (Amazon S3). Para obtener más información, consulte Ejemplo: cree la configuración de distribución para los discos de máquinas virtuales de salida desde AWS CLI.

Distribución de imágenes de contenedor

- Especifique el repositorio de ECR donde Generador de Imágenes almacena la imagen de salida en la región de distribución.

Puedes usar tu configuración de distribución de las siguientes maneras para enviar imágenes a las regiones, cuentas AWS Organizations y unidades organizativas (OUs) de destino una sola vez o con cada proceso de creación:

- Para entregar automáticamente las imágenes actualizadas a regiones, cuentas y organizaciones específicas y OUs utilizar la configuración de distribución con una canalización de Image Builder que se ejecute según un cronograma.
- Para crear una imagen nueva y entregarla a las regiones, cuentas y organizaciones especificadas OUs, utilice la configuración de distribución con una canalización de Image Builder que ejecute una vez desde la consola de Image Builder, utilizando Ejecutar canalización en el menú Acciones.
- Para crear una imagen nueva y entregarla a las regiones, cuentas y organizaciones especificadas y OUs, utilice la configuración de distribución con la siguiente acción de API o el comando Image Builder en AWS CLI:
  - La acción CreateImage de la API de Generador de Imágenes.
  - El comando create-image en la AWS CLI.
- Exportar discos de imágenes de máquinas virtuales (VM) a buckets S3 en las regiones de destino como parte del proceso habitual de creación de imágenes.

> **Tip**
>
> Cuando tenga muchos recursos del mismo tipo, el etiquetado puede ayudarle a identificar un recurso específico en función de las etiquetas que le haya asignado. Para obtener más

información sobre cómo etiquetar los recursos mediante los comandos de Image Builder de AWS CLI, consulte la Etiquetar recursos sección de esta guía.

Contenido

- Enumeración y visualización de los detalles de la configuración de la distribución
- Creación y actualización de ajustes de distribución de AMI
- Creación y actualización de los ajustes de la distribución de las imágenes de los contenedores
- Configuración de la distribución entre cuentas de AMI utilizando Generador de Imágenes
- Configurar la distribución de AMI con una plantilla de EC2 lanzamiento
- Utilice capacidades de distribución de AMI mejoradas

Enumeración y visualización de los detalles de la configuración de la distribución

En esta sección se describen las distintas formas de encontrar información y ver los detalles de la configuración de distribución de EC2 Image Builder.

Detalles de la configuración de la distribución

- Enumere las configuraciones de distribución de la consola
- Vea los detalles de la configuración de la distribución desde la consola
- Enumere las distribuciones del AWS CLI
- Obtenga los detalles de la configuración de distribución en la AWS CLI

Enumere las configuraciones de distribución de la consola

Para ver una lista de los ajustes de la distribución creados en su cuenta en la consola de Generador de Imágenes, siga estos pasos:

1. Abra la consola EC2 de Image Builder en https://console.aws.amazon.com/imagebuilder/.
2. En el panel de navegación, seleccione Ajustes de distribución. Aquí se muestra una lista de los ajustes de distribución que se crean en su cuenta.
3. Para ver los detalles o crear una nueva configuración de la distribución, elija el enlace nombre de la configuración. Esto abre la vista detallada de los ajustes de la distribución.

> **Note**
>
> También puede seleccionar la casilla junto al nombre de la configuración, y, a continuación, elija Ver detalles.

Vea los detalles de la configuración de la distribución desde la consola

Para ver los detalles de una configuración de la distribución específica mediante la consola Generador de Imágenes, seleccione la configuración que desee revisar siguiendo los pasos que se describen en Enumere las configuraciones de distribución de la consola.

En la página de detalles de la distribución, puede:

- Elimine una configuración de distribución. Para obtener más información sobre cómo eliminar recursos en Generador de Imágenes, consulte Eliminación de recursos de Generador de imágenes obsoletos o no utilizados.
- Edite detalles de la distribución.

Enumere las distribuciones del AWS CLI

El siguiente ejemplo muestra cómo utilizar el list-distribution-configurations comando AWS CLI para enumerar todas las distribuciones.

```
aws imagebuilder list-distribution-configurations
```

Obtenga los detalles de la configuración de distribución en la AWS CLI

El siguiente ejemplo muestra cómo utilizar el get-distribution-configuration comando AWS CLI para obtener los detalles de una configuración de distribución especificando su nombre de recurso de Amazon (ARN).

```
aws imagebuilder get-distribution-configuration --distribution-configuration-arn
 arn:aws:imagebuilder:us-west-2:123456789012:distribution-configuration/my-example-
distribution-configuration
```

Creación y actualización de ajustes de distribución de AMI

En esta sección se describe la creación y actualización de los ajustes de distribución para una AMI de Generador de Imágenes.

Contenido

- Requisitos previos de distribución de AMI
- Crear una configuración de distribución de AMI
- Actualizar una configuración de distribución de AMI
- Ejemplo: habilite EC2 Fast Launch con una plantilla de lanzamiento como salida AMIs
- Ejemplo: cree la configuración de distribución para los discos de máquinas virtuales de salida desde AWS CLI

Requisitos previos de distribución de AMI

Algunas configuraciones de distribución tienen requisitos previos, como los siguientes:

Temas

- Requisitos previos para los parámetros de salida del SSM
- Requisitos previos para un lanzamiento EC2 rápido

Requisitos previos para los parámetros de salida del SSM

Antes de crear una nueva configuración de distribución de AMI que establezca un AWS Systems Manager parámetro de almacén de parámetros (parámetro SSM), asegúrese de cumplir los siguientes requisitos previos.

Rol de ejecución

Al crear una canalización o utilizar el comando create-image en AWS CLI, solo puede especificar una función de ejecución de Image Builder. Si ha definido una función de ejecución del flujo de trabajo de Image Builder, añadiría cualquier permiso de función adicional a esa función. De lo contrario, crearía un nuevo rol personalizado que incluya los permisos necesarios.

- Para almacenar el ID de AMI de salida en un parámetro de SSM durante la distribución, debe especificar la `ssm:PutParameter` acción en su rol de ejecución de Image Builder, con el parámetro listado como recurso.

- Al configurar el tipo de datos del parámetro para indicar `AWS EC2 Image` a Systems Manager que valide el valor del parámetro como un ID de AMI, también debe añadir la `ec2:DescribeImages` acción.

Requisitos previos para un lanzamiento EC2 rápido

Antes de crear una nueva configuración de distribución para EC2 Fast Launch para Windows AMIs, asegúrese de cumplir los siguientes requisitos previos.

- Si proporciona una plantilla de lanzamiento personalizada al configurar EC2 Fast Launch, el servicio utiliza la VPC y otros ajustes de configuración que haya definido en la plantilla de lanzamiento. Para obtener más información, consulte Usar una plantilla de lanzamiento al configurar EC2 Fast Launch.
- Si no utiliza una plantilla de lanzamiento personalizada para configurar los ajustes, debe adjuntar la EC2FastLaunchFullAccesspolítica al rol de IAM que Image Builder utiliza para crear la imagen. Al crear una canalización o utilizar el comando create-image en AWS CLI, solo puede especificar una función de ejecución de Image Builder. Si ha definido una función de ejecución del flujo de trabajo de Image Builder, añadiría cualquier permiso de función adicional a esa función. De lo contrario, crearía un nuevo rol personalizado que incluya los permisos necesarios.

  A continuación, cuando Image Builder copia la imagen, EC2 Fast Launch crea automáticamente una CloudFormation pila con los siguientes recursos en su interior Cuenta de AWS.

  - Una nube privada virtual (VPC)
  - Subredes privadas en varias zonas de disponibilidad
  - Una plantilla de lanzamiento configurada con la versión 2 (IMDSv2) del Servicio de Metadatos de Instancia
  - Un grupo de seguridad sin reglas entrantes ni salientes

> **Note**
>
> Image Builder no admite la distribución entre cuentas si EC2 Fast Launch está preactivado. AMIs EC2 Fast Launch debe estar activado desde la cuenta de destino.

Crear una configuración de distribución de AMI

Las configuraciones de distribución incluyen el nombre de la AMI de salida, la configuración regional específica para el cifrado, los permisos de lanzamiento y Cuentas de AWS las organizaciones y unidades organizativas (OUs) que pueden lanzar la AMI de salida y las configuraciones de licencia.

Una configuración de distribución le permite especificar el nombre y la descripción de la AMI de salida, autorizar Cuentas de AWS a otras personas a lanzar la AMI, copiar la AMI a otras cuentas y replicar la AMI en otras AWS regiones. También le permite exportar la AMI a Amazon Simple Storage Service (Amazon S3) o EC2 configurar Fast Launch para Windows de salida. AMIs Para hacer una AMI pública, configure las cuentas autorizadas de permiso de lanzamiento en `all`. Consulte los ejemplos de cómo hacer pública una AMI en EC2 ModifyImageAttribute.

Console

Siga estos pasos para crear una nueva configuración de distribución de AMI en Consola de administración de AWS:

1. Abra la consola EC2 de Image Builder en https://console.aws.amazon.com/imagebuilder/.
2. En el panel de navegación, seleccione Ajustes de distribución. Aquí se muestra una lista de los ajustes de distribución que se crean en su cuenta.
3. Seleccione Crear ajustes de la distribución cerca de la parte superior del panel de Ajustes de distribución.
4. En la sección Tipo de imagen, elija Imagen de máquina de Amazon (AMI) como tipo de salida.
5. En la sección General, introduzca un nombre para la configuración de distribución y una descripción opcional.
6. En la sección Ajustes de la región, introduzca los siguientes detalles para cada región en la que distribuya su AMI:

    a. La AMI se distribuye a la región actual (región 1) de forma predeterminada. La región 1 es el origen de la distribución. Algunos ajustes de la región 1 no están abiertos para editarlos. Para cualquier región que añada, puede elegir una región de la lista desplegable de regiones.

    La clave Kms identifica la AWS KMS key que se utiliza para cifrar los volúmenes de EBS de la imagen en la región de destino. Es importante tener en cuenta que esto no se aplica a la AMI original que la compilación crea en su cuenta en la región de origen

(región 1). El cifrado que se ejecuta durante la fase de distribución de la creación es solo para las imágenes que se distribuyen a otras cuentas o regiones.

Para cifrar los volúmenes de EBS para la AMI que se creó en la región de origen para su cuenta, debe configurar la clave KMS en la asignación de dispositivos de bloques de recetas de imagen (almacenamiento (volúmenes) en la consola).

Generador de Imágenes copia la AMI en las cuentas de destino que especifique para la región.

> **Requisito previo**
>
> Para copiar una imagen entre cuentas, debe crear el rol `EC2ImageBuilderDistributionCrossAccountRole` en todas las cuentas de destino de distribución y adjuntar la política administrada Política Ec2ImageBuilderCrossAccountDistributionAccess al rol.

El nombre de la AMI de salida es opcional. Si proporciona un nombre, el nombre de la AMI de salida final incluirá una marca de tiempo adjunta que indica cuando se creó la AMI. Si no especifica un nombre, Generador de Imágenes añade la marca de tiempo de creación al nombre de la receta. Esto garantiza nombres de AMI únicos para cada creación.

i. Con el uso compartido de AMI, puede conceder acceso a AWS directores específicos para lanzar instancias desde su AMI. Si amplía la sección Uso compartido de AMI, puede introducir los siguientes detalles:

   - Permisos de lanzamiento: seleccione Privado si quiere mantener la privacidad de su AMI y permitir el acceso a AWS directores específicos para lanzar una instancia desde su AMI privada. Seleccione Público si quiere que su AMI sea pública. Cualquier AWS director puede lanzar una instancia desde su AMI pública.
   - Principales: puede conceder acceso a los siguientes tipos de AWS principales para lanzar instancias:
     - AWS cuenta: otorga acceso a una cuenta específica AWS
     - Unidad organizativa (OU): permite el acceso a una OU y a todas sus entidades secundarias. Las entidades secundarias incluyen AWS cuentas OUs y.

- Organización: conceda acceso a su AWS Organizations entidad secundaria y a todas sus entidades secundarias. Las entidades secundarias incluyen AWS las cuentas OUs y.

  En primer lugar, seleccione el tipo de entidad principal. A continuación, introduzca el ID de la entidad principal de AWS a la que desea conceder acceso en el cuadro situado a la derecha de la lista desplegable. Puede introducir varios IDs tipos diferentes.

ii. Puede ampliar la sección Configuración de licencias para adjuntar las configuraciones de licencia creadas con AWS License Manager las imágenes de Image Builder. Las configuraciones de la licencia contienen reglas de asignación de licencias que se basan en las condiciones de los contratos de su empresa. Generador de Imágenes incluye automáticamente las configuraciones de licencias asociadas a la AMI base.

iii. Puede ampliar la sección de configuración de plantillas de lanzamiento para especificar una plantilla de EC2 lanzamiento que se utilizará para lanzar instancias desde la AMI que cree.

  Si utiliza una plantilla de EC2 lanzamiento, puede indicar a Image Builder que cree una nueva versión de la plantilla de lanzamiento que incluya el ID de AMI más reciente una vez finalizada la compilación. Para actualizar la plantilla de lanzamiento, configure los ajustes de la siguiente manera:

  - Nombre de la plantilla de lanzamiento: seleccione el nombre de la plantilla de lanzamiento que desea que Generador de Imágenes actualice.
  - Establecer la versión predeterminada: seleccione esta casilla de verificación para actualizar la versión predeterminada de la plantilla de lanzamiento a la versión nueva.

  Para añadir otra configuración de la plantilla de lanzamiento, elija Añadir configuración de la plantilla de lanzamiento. Puede disponer de hasta cinco configuraciones de plantillas de lanzamiento por región.

iv. Puede ampliar la sección de configuraciones de parámetros de SSM para configurar un parámetro de SSM que almacenará el ID de AMI de salida de la imagen que se distribuye a la región de destino. Si lo desea, puede especificar una cuenta de distribución en la región.

Nombre del parámetro: introduzca el nombre del parámetro. Por ejemplo, `/output/image/param`.

Tipo de datos: mantenga el valor predeterminado (`AWS EC2 Image`). Esto le indica a Systems Manager que valide el valor del parámetro para asegurarse de que es un ID de AMI válido.

b. Para añadir los ajustes de la distribución para otra región, seleccione Añadir región.

7. Cuando haya terminado, elija Crear ajustes.

AWS CLI

El siguiente ejemplo muestra cómo utilizar el comando create-distribution-configuration para crear una nueva configuración de distribución para la AMI mediante la AWS CLI.

1. Creación de un archivo JSON de entrada de la CLI

   Utilice una herramienta de edición de archivos para crear un archivo JSON con las claves que se muestran en uno de los siguientes ejemplos y valores que sean válidos para su entorno. Estos ejemplos definen quéCuentas de AWS unidades organizativas (OUs) tienen permiso para lanzar la AMI que distribuyes en las regiones especificadas.AWS Organizations Asigne un nombre al archivo `create-ami-distribution-configuration.json` para su uso en el paso siguiente:

   Ejemplo 1: Distribuir a Cuentas de AWS

   En este ejemplo, se distribuye una AMI a dos regiones y se especifica quéCuentas de AWS tienen permisos de lanzamiento en cada región.

```
{
 "name": "MyExampleAccountDistribution",
 "description": "Copies AMI to eu-west-1, and specifies accounts that can launch
 instances in each Region.",
 "distributions": [
  {
   "region": "us-west-2",
   "amiDistributionConfiguration": {
    "name": "Name {{imagebuilder:buildDate}}",
    "description": "An example image name with parameter references",
    "amiTags": {
```

```
      "KeyName": "Some Value"
     },
     "launchPermission": {
      "userIds": [
       "987654321012"
      ]
     }
    }
   },
   {
    "region": "eu-west-1",
    "amiDistributionConfiguration": {
     "name": "My {{imagebuilder:buildVersion}} image {{imagebuilder:buildDate}}",
     "amiTags": {
      "KeyName": "Some value"
     },
     "launchPermission": {
      "userIds": [
       "100000000001"
      ]
     }
    }
   }
  ]
}
```

Ejemplo 2: Distribuir a Organizations y OUs

En este ejemplo, se distribuye una AMI a la región de origen y se especifican los permisos de lanzamiento de la organización y la unidad organizativa.

```
{
 "name": "MyExampleAWSOrganizationDistribution",
 "description": "Shares AMI with the Organization and OU",
 "distributions": [
  {
   "region": "us-west-2",
   "amiDistributionConfiguration": {
    "name": "Name {{ imagebuilder:buildDate }}",
    "launchPermission": {
     "organizationArns": [
      "arn:aws:organizations::123456789012:organization/o-myorganization123"
     ],
```

```
     "organizationalUnitArns": [
      "arn:aws:organizations::123456789012:ou/o-123example/ou-1234-
myorganizationalunit"
     ]
    }
   }
  }
 ]
}
```

Ejemplo 3: almacenar el ID de AMI de salida en un parámetro SSM

En este ejemplo, se almacena el ID de AMI de salida en un AWS Systems Manager parámetro del almacén de parámetros de la región de distribución.

```
{
 "name": "SSMParameterOutputAMI",
 "description": "Updates an SSM parameter with the output AMI ID for the
 distribution.",
 "distributions": [
  {
   "region": "us-west-2",
   "amiDistributionConfiguration": {
    "name": "Name {{ imagebuilder:buildDate }}"
   },
   "ssmParameterConfigurations": [
    {
     "amiAccountId": "111122223333",
     "parameterName": "/output/image/param",
     "dataType": "aws:ec2:image"
    }
   ]
  }
 ]
}
```

2. Ejecute el siguiente comando utilizando el archivo que creó como entrada.

```
aws imagebuilder create-distribution-configuration --cli-input-json
 file://create-ami-distribution-configuration.json
```

> **Note**
>
> - Debe incluir la notación `file://` al principio de la ruta del archivo JSON.
> - La ruta del archivo JSON debe seguir la convención apropiada para el sistema operativo base donde se está ejecutando el comando. Por ejemplo, Windows utiliza la barra diagonal inversa (\) para referirse a la ruta de directorios, mientras que Linux y macOS usan la barra diagonal (/).

Para obtener información más detallada, consulte create-distribution-configuration en la referencia de comandos de la AWS CLI.

Actualizar una configuración de distribución de AMI

Puede cambiar la configuración de distribución de la AMI. Sin embargo, los cambios que realice no se aplican a ningún recurso que Generador de Imágenes ya haya distribuido. Por ejemplo, si ha distribuido una AMI en una región y luego la elimina de la distribución, la AMI que ya estaba distribuida permanece en esa región hasta que la elimine manualmente.

Consola de administración de AWS

Siga estos pasos para configurar una distribución de AMI en Consola de administración de AWS:

1. Abra la consola EC2 de Image Builder en https://console.aws.amazon.com/imagebuilder/.
2. En el panel de navegación, seleccione Ajustes de distribución. Aquí se muestra una lista de los ajustes de distribución que se crean en su cuenta.
3. Para ver los detalles o actualizar una configuración de distribución, elija el enlace nombre de la configuración. Esto abre la vista detallada de los ajustes de la distribución.

> **Note**
>
> También puede seleccionar la casilla junto al nombre de la configuración, y, a continuación, elija Ver detalles.

4. Para editar la configuración de distribución, elija Editar en la esquina superior derecha de la sección Detalles de distribución. Algunos campos están bloqueados, como el nombre de la

configuración de distribución y la región predeterminada que se muestra como región 1. Para obtener más información acerca de los ajustes de la configuración de distribución, consulte Crear una configuración de distribución de AMI.

5. Cuando haya terminado, elija Guardar cambios.

AWS CLI

El siguiente ejemplo muestra cómo utilizar el comando update-distribution-configuration para actualizar los ajustes de la distribución de la AMI mediante la AWS CLI.

1. Creación de un archivo JSON de entrada de la CLI

   Utilice una herramienta de edición de archivos para crear un archivo JSON con las claves que se muestran en el siguiente ejemplo y valores que sean válidos para su entorno. En este ejemplo, se utiliza un archivo con el nombre `update-ami-distribution-configuration.json`.

```
{
 "distributionConfigurationArn": "arn:aws:imagebuilder:us-
west-2:123456789012:distribution-configuration/update-ami-distribution-
configuration.json",
 "description": "Copies AMI to eu-west-2, and specifies accounts that can launch
 instances in each Region.",
 "distributions": [
   {
   "region": "us-west-2",
   "amiDistributionConfiguration": {
    "name": "Name {{imagebuilder:buildDate}}",
    "description": "An example image name with parameter references",
    "launchPermissions": {
     "userIds": [
      "987654321012"
     ]
    }
   }
  },
  {
   "region": "eu-west-2",
   "amiDistributionConfiguration": {
    "name": "My {{imagebuilder:buildVersion}} image {{imagebuilder:buildDate}}",
    "tags": {
```

```
      "KeyName": "Some value"
     },
     "launchPermissions": {
      "userIds": [
       "100000000001"
      ]
     }
    }
   }
  ]
}
```

2. Ejecute el siguiente comando utilizando el archivo que creó como entrada.

```
aws imagebuilder update-distribution-configuration --cli-input-json
 file://update-ami-distribution-configuration.json
```

> **Note**
>
> - Debe incluir la notación `file://` al principio de la ruta del archivo JSON.
> - La ruta del archivo JSON debe seguir la convención apropiada para el sistema operativo base donde se está ejecutando el comando. Por ejemplo, Windows utiliza la barra diagonal inversa (\) para referirse a la ruta de directorios, mientras que Linux y macOS usan la barra diagonal (/).

Para obtener información más detallada, consulte update-distribution-configuration en la referencia de comandos de la AWS CLI. Para actualizar las etiquetas de su recurso de configuración de distribución, consulte la sección Etiquetar recursos.

Ejemplo: habilite EC2 Fast Launch con una plantilla de lanzamiento como salida AMIs

El siguiente ejemplo muestra cómo utilizar el create-distribution-configuration comando con una plantilla de lanzamiento para crear ajustes de distribución que tengan EC2 Fast Launch configurado para la AMI, desde AWS CLI.

Para configurar los ajustes de EC2 Fast Launch sin una plantilla de lanzamiento, asegúrese de cumplir con todos los EC2 Requisitos previos de inicio rápido requisitos antes de crear la configuración de distribución.

1. Creación de un archivo JSON de entrada de la CLI

   Utilice una herramienta de edición de archivos para crear un archivo JSON con claves, como se muestra en el siguiente ejemplo, además de valores que sean válidos para su entorno.

   En este ejemplo, se lanzan instancias para todos sus recursos de destino de forma simultánea, ya que el número máximo de lanzamientos en paralelo es mayor que el recuento de recursos de destino. El nombre de este archivo es `ami-dist-config-win-fast-launch.json` en el ejemplo de comando que se muestra en el siguiente paso.

```
{
"name": "WinFastLaunchDistribution",
"description": "An example of Windows AMI EC2 Fast Launch settings in the
 distribution configuration.",
"distributions": [
 {
  "region": "us-west-2",
  "amiDistributionConfiguration": {
   "name": "Name {{imagebuilder:buildDate}}",
   "description": "Includes Windows AMI EC2 Fast Launch settings.",
   "amiTags": {
    "KeyName": "Some Value"
   }
  },
  "fastLaunchConfigurations": [{
   "enabled": true,
   "snapshotConfiguration": {
    "targetResourceCount": 5
   },
   "maxParallelLaunches": 6,
   "launchTemplate": {
    "launchTemplateId": "lt-0ab1234c56d789012",
    "launchTemplateVersion": "1"
    }
  }],
  "launchTemplateConfigurations": [{
       "launchTemplateId": "lt-0ab1234c56d789012",
       "setDefaultVersion": true
    }]
 }]
}
```

> **Note**
>
> Puede especificar el `launchTemplateName` en lugar del `launchTemplateId` en la sección `launchTemplate`, pero no puede especificar tanto el nombre como el identificador.

2. Ejecute el siguiente comando utilizando el archivo que creó como entrada.

```
aws imagebuilder create-distribution-configuration --cli-input-json file://ami-dist-config-win-fast-launch.json
```

> **Note**
>
> - Debe incluir la notación `file://` al principio de la ruta del archivo JSON.
> - La ruta del archivo JSON debe seguir la convención apropiada para el sistema operativo base donde se está ejecutando el comando. Por ejemplo, Windows utiliza la barra diagonal inversa (\) para referirse a la ruta de directorios, mientras que Linux y macOS usan la barra diagonal (/).

Para obtener información más detallada, consulte create-distribution-configuration en la referencia de comandos de la AWS CLI.

Ejemplo: cree la configuración de distribución para los discos de máquinas virtuales de salida desde AWS CLI

El siguiente ejemplo muestra cómo usar el comando create-distribution-configuration para crear ajustes de la distribución que exportarán los discos de imágenes de máquinas virtuales a Amazon S3 con cada creación de imágenes.

1. Creación de un archivo JSON de entrada de la CLI

   Puede simplificar el comando create-distribution-configuration que utiliza en la AWS CLI. Para ello, cree un archivo JSON que contenga toda la configuración de exportación que desee pasar al comando.

> **Note**
>
> La convención de nomenclatura de los valores de datos del archivo JSON sigue el patrón que se especifica para los parámetros de solicitud de operación de la API Image Builder. Para revisar los parámetros de la solicitud de operación de la API, consulte el CreateDistributionConfiguration comando en la Referencia de la API de EC2 Image Builder.
>
> Para proporcionar los valores de los datos como parámetros de la línea de comandos, consulte los nombres de los parámetros especificados en la referencia de comandos de la AWS CLI y utilice el comando create-distribution-configuration como opciones.

Este es un resumen de los parámetros que especificamos en el objeto `s3ExportConfiguration` de JSON para este ejemplo:

- roleName (cadena, obligatorio): el nombre del rol que concede a la máquina virtual el Import/Export permiso para exportar imágenes al bucket de S3.
- diskImageFormat(cadena, obligatorio): exporta la imagen de disco actualizada a uno de los siguientes formatos compatibles:
  - Virtual Hard Disk (VHD), que es compatible con los productos de virtualización Citrix Xen y Microsoft Hyper-V.
  - Disco de máquina virtual ESX optimizado para streaming (VMDK): compatible con las versiones 4, 5 y 6 de ESX VMware y VMware vSphere.
  - Crudo: formato en crudo.
- S3Bucket (cadena, obligatorio): el bucket S3 en el que se almacenan las imágenes de disco de salida de la máquina virtual.

Guarde el archivo como `export-vm-disks.json`. Usa el nombre del archivo en el comando create-distribution-configuration.

```
{
 "name": "example-distribution-configuration-with-vm-export",
 "description": "example",
 "distributions": [
  {
   "region": "us-west-2",
```

```
   "amiDistributionConfiguration": {
    "description": "example-with-vm-export"

   },
   "s3ExportConfiguration": {
    "roleName": "vmimport",
    "diskImageFormat": "RAW",
    "s3Bucket": "vm-bucket-export"
   }
  }],
 "clientToken": "abc123def4567ab"
}
```

2. Ejecute el siguiente comando utilizando el archivo que creó como entrada.

```
aws imagebuilder create-distribution-configuration --cli-input-json file://export-vm-disks.json
```

> **Note**
>
> - Debe incluir la notación `file://` al principio de la ruta del archivo JSON.
> - La ruta del archivo JSON debe seguir la convención apropiada para el sistema operativo base donde se está ejecutando el comando. Por ejemplo, Windows utiliza la barra diagonal inversa (\) para referirse a la ruta de directorios, mientras que Linux y macOS usan la barra diagonal (/).

Para obtener información más detallada, consulte create-distribution-configuration en la referencia de comandos de la AWS CLI.

Creación y actualización de los ajustes de la distribución de las imágenes de los contenedores

En esta sección se describe la creación y actualización de los ajustes de distribución de las imágenes de contenedores de Generador de Imágenes.

Contenido

- Cree ajustes de distribución para las imágenes de contenedor de Image Builder desde AWS CLI

- Actualice la configuración de distribución de la imagen de su contenedor desde AWS CLI

Cree ajustes de distribución para las imágenes de contenedor de Image Builder desde AWS CLI

Una configuración de distribución le permite especificar el nombre y la descripción de la imagen del contenedor de salida y replicar la imagen del contenedor en otras AWS regiones. También puede aplicar etiquetas independientes al recurso de configuración de distribución y a las imágenes del contenedor dentro de cada región.

1. Creación de un archivo JSON de entrada de la CLI

   Utilice su herramienta de edición de archivos favorita para crear un archivo JSON con las claves que se muestran en el siguiente ejemplo, además de valores que sean válidos para su entorno. En este ejemplo, se utiliza un archivo con el nombre `create-container-distribution-configuration.json`:

```
{
 "name": "distribution-configuration-name",
 "description": "Distributes container image to Amazon ECR repository in two
 regions.",
 "distributions": [
  {
   "region": "us-west-2",
   "containerDistributionConfiguration": {
    "description": "My test image.",
    "targetRepository": {
     "service": "ECR",
     "repositoryName": "testrepo"
    },
    "containerTags": ["west2", "image1"]
   }
  },
  {
   "region": "us-east-1",
   "containerDistributionConfiguration": {
    "description": "My test image.",
    "targetRepository": {
     "service": "ECR",
     "repositoryName": "testrepo"
    },
```

```
      "containerTags": ["east1", "imagedist"]
    }
   }
  ],
  "tags": {
     "DistributionConfigurationTestTagKey1":
  "DistributionConfigurationTestTagValue1",
     "DistributionConfigurationTestTagKey2":
  "DistributionConfigurationTestTagValue2"
  }
}
```

2. Ejecute el siguiente comando utilizando el archivo que creó como entrada.

```
aws imagebuilder create-distribution-configuration --cli-input-json file://create-container-distribution-configuration.json
```

> **Note**
>
> - Debe incluir la notación `file://` al principio de la ruta del archivo JSON.
> - La ruta del archivo JSON debe seguir la convención apropiada para el sistema operativo base donde se está ejecutando el comando. Por ejemplo, Windows utiliza la barra diagonal inversa (\) para referirse a la ruta de directorios, mientras que Linux y macOS usan la barra diagonal (/).

Para obtener información más detallada, consulte create-distribution-configuration en la referencia de comandos de la AWS CLI.

Actualice la configuración de distribución de la imagen de su contenedor desde AWS CLI

El siguiente ejemplo muestra cómo utilizar el comando update-distribution-configuration para actualizar los ajustes de la distribución de la imagen de contenedor mediante la AWS CLI. También puede actualizar las etiquetas de las imágenes del contenedor en cada región.

1. Creación de un archivo JSON de entrada de la CLI

   Utilice su herramienta de edición de archivos favorita para crear un archivo JSON con las claves que se muestran en el siguiente ejemplo, además de valores que sean válidos para su entorno. En este ejemplo, se utiliza un archivo con el nombre `update-container-distribution-configuration.json`:

```
{
 "distributionConfigurationArn": "arn:aws:imagebuilder:us-
west-2:123456789012:distribution-configuration/update-container-distribution-
configuration.json",
 "description": "Distributes container image to Amazon ECR repository in two
 regions.",
 "distributions": [
  {
   "region": "us-west-2",
   "containerDistributionConfiguration": {
    "description": "My test image.",
    "targetRepository": {
     "service": "ECR",
     "repositoryName": "testrepo"
    },
    "containerTags": ["west2", "image1"]
   }
  },
  {
   "region": "us-east-2",
   "containerDistributionConfiguration": {
    "description": "My test image.",
    "targetRepository": {
     "service": "ECR",
     "repositoryName": "testrepo"
    },
      "containerTags": ["east2", "imagedist"]
   }
  }
 ]
}
```

2. Ejecute el siguiente comando utilizando el archivo que creó como entrada:

```
aws imagebuilder update-distribution-configuration --cli-input-json file://update-container-distribution-configuration.json
```

> **Note**
>
> - Debe incluir la notación `file://` al principio de la ruta del archivo JSON.
> - La ruta del archivo JSON debe seguir la convención apropiada para el sistema operativo base donde se está ejecutando el comando. Por ejemplo, Windows utiliza la barra diagonal inversa (\) para referirse a la ruta de directorios, mientras que Linux y macOS usan la barra diagonal (/).

Para obtener información más detallada, consulte update-distribution-configuration en la referencia de comandos de la AWS CLI. Para actualizar las etiquetas de su recurso de configuración de distribución, consulte la sección Etiquetar recursos.

Configuración de la distribución entre cuentas de AMI utilizando Generador de Imágenes

En esta sección se describe cómo configurar los ajustes de distribución para entregar una AMI de Generador de Imágenes a otras cuentas que especifique.

La cuenta de destino puede entonces lanzar o modificar la AMI, según sea necesario.

> **Note**
>
> AWS CLI En los ejemplos de comandos de esta sección se supone que ha creado previamente archivos JSON de configuración de infraestructura y recetas de imagen. Para crear el archivo JSON para una receta de imagen, consulte Cree una receta con imágenes con el AWS CLI. Para crear el archivo JSON para una configuración de infraestructura, consulte Crear una configuración de infraestructura.

Requisitos previos para la distribución entre cuentas de AMI

Para garantizar que las cuentas de destino puedan lanzar instancias correctamente desde su imagen de Generador de Imágenes, debe configurar los permisos adecuados para todas las cuentas de destino de todas las regiones.

Si cifra su AMI mediante AWS Key Management Service(AWS KMS), debe configurar una AWS KMS key para su cuenta que se utilice para cifrar la nueva imagen.

Cuando Image Builder realiza una distribución entre cuentas para el cifrado AMIs, la imagen de la cuenta de origen se descifra y se envía a la región de destino, donde se vuelve a cifrar con la clave designada para esa región. Como Generador de Imágenes actúa en nombre de la cuenta de destino y utiliza un rol de IAM que usted crea en la región de destino, esa cuenta debe tener acceso a las claves de la región de origen y de destino.

Clave de cifrado

Los requisitos previos siguientes si su imagen se cifra utilizando AWS KMS. Los requisitos previos de IAM se describen en la siguiente sección.

Requisitos de cuenta de origen

- Cree una clave KMS en su cuenta en todas las regiones en las que cree y distribuya su AMI. También puede utilizar una clave que ya exista.
- Actualiza la política de claves para todas esas claves para permitir que las cuentas de destino usen su clave.

Requisitos de cuenta de destino

- Añada una política en línea `EC2ImageBuilderDistributionCrossAccountRole` que permita al rol realizar las acciones necesarias para distribuir una AMI cifrada. Para ver los pasos de configuración de IAM, consulte la sección de requisitos previos Políticas de IAM.

Para obtener más información sobre el acceso entre cuentas AWS KMS, consulte Permitir que los usuarios de otras cuentas usen una clave KMS en la Guía para desarrolladores.AWS Key Management Service

Especifique la clave de cifrado en la receta de la imagen, de la siguiente manera:

- Si utiliza la consola Generador de Imágenes, elija la clave de cifrado en la lista desplegable Cifrado (alias KMS) de la sección Almacenamiento (volúmenes) de su receta.
- Si utilizas la acción de la CreateImageRecipe API o el create-image-recipe comando de la AWS CLI, configura tu clave en la `ebs` sección situada debajo `blockDeviceMappings` de la entrada de JSON.

  En el siguiente fragmento de código JSON, se muestran los ajustes de cifrado de una receta de imagen. Además de proporcionar la clave de cifrado, también debe establecer el indicador `encrypted` como `true`.

```
{
 ...
 "blockDeviceMappings": [
 {
  "deviceName": "Example root volume",
  "ebs": {
   "deleteOnTermination": true,
   "encrypted": true,
   "iops": 100,
   "kmsKeyId": "image-owner-key-id",
   ...
  },
  ...
 }],
 ...
}
```

Políticas de IAM

Para configurar los permisos de distribución entre cuentas en AWS Identity and Access Management(IAM), sigue estos pasos:

1. Para usar Image Builder AMIs que se distribuyen entre las cuentas, el propietario de la cuenta de destino debe crear un nuevo rol de IAM en su cuenta denominado`EC2ImageBuilderDistributionCrossAccountRole`.
2. Deben asociar Política Ec2ImageBuilderCrossAccountDistributionAccess al rol para permitir la distribución entre cuentas. Para obtener más información sobre políticas administradas, consulte Políticas administradas y políticas insertadas en la AWS Identity and Access Management Guía del usuario de IAM.

3. Compruebe que el ID de la cuenta de origen se haya añadido a la política de confianza asociada al rol de IAM de la cuenta de destino. El siguiente ejemplo muestra una política de confianza en la cuenta de destino que especifica el ID de cuenta de la cuenta de origen.

   JSON

   ```
   {
       "Version":"2012-10-17",
       "Statement": [{
           "Effect": "Allow",
           "Principal": {
               "AWS": "arn:aws:iam::444455556666:root"
           },
           "Action": "sts:AssumeRole"
       }]
   }
   ```

   Para obtener más información sobre las políticas de confianza, consulte Políticas basadas en recursos en laGuía del usuario de AWS Identity and Access Management.

4. Si la AMI que distribuye está cifrada, el propietario de la cuenta de destino debe añadir la siguiente política en línea al `EC2ImageBuilderDistributionCrossAccountRole` en su cuenta para poder usar sus claves de KMS. La sección `Principal` contiene su número de cuenta. Esto permite a Image Builder actuar en su nombre cuando se utiliza AWS KMS para cifrar y descifrar la AMI con las claves adecuadas para cada región.

   JSON

   ```
   {
    "Version":"2012-10-17",
    "Statement": [
     {
      "Sid": "AllowRoleToPerformKMSOperationsOnBehalfOfTheDestinationAccount",
      "Effect": "Allow",
      "Action": [
       "kms:Encrypt",
       "kms:Decrypt",
       "kms:ReEncrypt*",
       "kms:GenerateDataKey*",
   ```

```
    "kms:DescribeKey",
    "kms:CreateGrant",
    "kms:ListGrants",
    "kms:RevokeGrant"
   ],
   "Resource": "*"
  }
 ]
}
```

Para obtener más información acerca de las políticas insertadas, consulte Políticas insertadas en la AWS Identity and Access Management Guía del usuario.

5. Si utilizas `launchTemplateConfigurations` para especificar una plantilla de EC2 lanzamiento de Amazon, también debes añadir la siguiente política a tu `EC2ImageBuilderDistributionCrossAccountRole` cuenta de destino.

   JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateLaunchTemplateVersion",
                "ec2:ModifyLaunchTemplate"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/CreatedBy": "EC2 Image Builder"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeLaunchTemplates"
            ],
            "Resource": "*"
```

```
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateTags"
            ],
            "Resource": "arn:aws:ec2:*:*:launch-template/*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/CreatedBy": "EC2 Image Builder"
                }
            }
        }
    ]
}
```

6. Si utiliza un AWS Systems Manager parámetro de almacén de parámetros para almacenar el ID de AMI de la AMI de salida de la cuenta de distribución y la región, debe añadir la siguiente política a la suya `EC2ImageBuilderDistributionCrossAccountRole` en cada cuenta de destino.

   JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:PutParameter"
            ],
            "Resource": "arn:aws:ssm:*:111122223333:parameter/ImageBuilder-*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeImages"
            ],
            "Resource": "*"
        }
    ]
}
```

Límites para la distribución entre cuentas

Existen algunas limitaciones a la hora de distribuir imágenes de Generador de Imágenes entre cuentas:

- La cuenta de destino está limitada a 50 copias simultáneas de la AMI para cada región de destino.
- Si desea copiar una AMI de virtualización paravirtual (PV) a otra región, la región de destino debe admitir la virtualización AMIs fotovoltaica. Para obtener más información, consulte Tipos de virtualización de AMI de Linux.
- No puede crear una copia sin cifrar de una instantánea cifrada. Si no especifica una clave gestionada por el cliente AWS Key Management Service(AWS KMS) para el parámetro `KmsKeyId`, Generador de Imágenes utilizará la clave predeterminada para Amazon Elastic Block Store (Amazon EBS). Para obtener más información, consulte Cifrado de Amazon EBS en la Guía del usuario de Amazon Elastic Compute Cloud.

Para obtener más información, consulte CreateDistributionConfigurationla Referencia de la API de EC2 Image Builder.

Configurar la distribución entre cuentas para una AMI de Generador de imágenes de la consola

En esta sección se describe cómo crear y configurar los ajustes de distribución para la distribución multicuenta de su Image Builder AMIs mediante.Consola de administración de AWS La configuración de la distribución entre cuentas requiere permisos de IAM específicos. Debe completar el formulario Requisitos previos para la distribución entre cuentas de AMI de esta sección antes de continuar.

Para crear los ajustes de distribución mediante la consola Generador de Imágenes, siga estos pasos:

1. Abra la consola EC2 de Image Builder en https://console.aws.amazon.com/imagebuilder/.
2. En el panel de navegación, seleccione Ajustes de distribución. Aquí se muestra una lista de los ajustes de distribución que se crean en su cuenta.
3. En la parte superior de la página Ajustes de distribución, seleccione Crear ajustes de distribución. Esto le llevará a la página Crear los ajustes de distribución.
4. En la sección Tipo de imagen, elija Imagen de máquina de Amazon (AMI) como tipo de salida. Esta es la configuración predeterminada.
5. En la sección General, ingrese el nombre del recurso de ajustes de distribución que desea crear (obligatorio).

6. En la sección Ajustes de región, introduzca un ID de cuenta de 12 dígitos al que desee distribuir su AMI en las cuentas de Target de la región seleccionada y pulse Entrar. De este modo, se comprueba que el formato es correcto y, a continuación, se muestra el ID de cuenta que ha introducido debajo del cuadro. Repite el proceso para añadir más cuentas.

   Para eliminar una cuenta que has introducido, seleccione la X que aparece a la derecha del ID de la cuenta.

   Introduzca el nombre de la AMI de salida para cada región.

7. Continúe especificando cualquier configuración adicional que necesite y seleccione Crear configuración para crear su nuevo recurso de ajustes de distribución.

Configurar la distribución entre cuentas de una AMI de Image Builder desde AWS CLI

En esta sección se describe cómo configurar un archivo de ajustes de distribución y cómo utilizar el create-image comando del AWS CLI para crear y distribuir una AMI de Image Builder en todas las cuentas.

La configuración de la distribución entre cuentas requiere permisos de IAM específicos. Debe completar el Requisitos previos para la distribución entre cuentas de AMI de esta sección antes de ejecutar el comando create-image.

1. Configuración de un archivo de ajustes de distribución

   Antes de utilizar el create-image comando AWS CLI para crear una AMI de Image Builder que se distribuya a otra cuenta, debe crear una estructura `DistributionConfiguration` JSON que especifique la cuenta de destino IDs en la `AmiDistributionConfiguration` configuración. Tiene que especificar al menos una `AmiDistributionConfiguration` en la región de origen.

   El siguiente archivo de ejemplo, denominado `create-distribution-configuration.json`, muestra la configuración para la distribución de imágenes entre cuentas en la región de origen.

```
{
 "name": "cross-account-distribution-example",
 "description": "Cross Account Distribution Configuration Example",
 "distributions": [
  {
```

```
   "amiDistributionConfiguration": {
    "targetAccountIds": ["123456789012", "987654321098"],
    "name": "Name {{ imagebuilder:buildDate }}",
    "description": "ImageCopy Ami Copy Configuration"
   },
   "region": "us-west-2"
  }
 ]
}
```

2. Creación de los ajustes de distribución

   Para crear un recurso de configuración de distribución de Image Builder mediante el create-distribution-configurationcomando del AWS CLI, proporcione los siguientes parámetros en el comando:

   - Ingrese el nombre de la distribución en el parámetro `--name`.
   - Adjunte el archivo JSON de configuración de distribución que creó en el parámetro `--cli-input-json`.

```
aws imagebuilder create-distribution-configuration --name my distribution name --
cli-input-json file://create-distribution-configuration.json
```

> **Note**
>
> - Debe incluir la notación `file://` al principio de la ruta del archivo JSON.
> - La ruta del archivo JSON debe seguir la convención apropiada para el sistema operativo base donde se está ejecutando el comando. Por ejemplo, Windows utiliza la barra diagonal inversa (\) para referirse a la ruta de directorios, mientras que Linux y macOS usan la barra diagonal (/).

También puede proporcionar JSON directamente en el comando, mediante el parámetro `--distributions`.

Configurar la distribución de AMI con una plantilla de EC2 lanzamiento

Para garantizar una experiencia de lanzamiento coherente para su AMI de Image Builder en las cuentas y regiones de destino, puede especificar una plantilla de EC2 lanzamiento de Amazon en su configuración de distribución mediante`launchTemplateConfigurations`. Cuando las `launchTemplateConfigurations` están presentes durante el proceso de distribución, Generador de Imágenes crea una nueva versión de la plantilla de lanzamiento que incluye todos los ajustes originales de la plantilla y el nuevo ID de AMI de la compilación. Para obtener más información sobre el lanzamiento de una EC2 instancia mediante una plantilla de lanzamiento, consulte uno de los siguientes enlaces, en función del sistema operativo de destino.

- Lanzamiento de una instancia de Linux desde una plantilla de lanzamiento
- Lanzamiento de una instancia de Windows desde una plantilla de lanzamiento

> **Note**
>
> Si incluye una plantilla de lanzamiento para habilitar el lanzamiento rápido de Windows en la imagen, la plantilla de lanzamiento debe incluir la siguiente etiqueta para que Generador de imágenes pueda habilitar el lanzamiento rápido de Windows en su nombre.
> `CreatedBy: EC2 Image Builder`

Agregue una plantilla de EC2 lanzamiento a la configuración de distribución de AMI desde la consola

Para proporcionar una plantilla de lanzamiento con la AMI de salida, siga estos pasos en la consola:

1. Abra la consola EC2 de Image Builder en https://console.aws.amazon.com/imagebuilder/.
2. En el panel de navegación, seleccione Ajustes de distribución. Aquí se muestra una lista de los ajustes de distribución que se crean en su cuenta.
3. En la parte superior de la página Ajustes de distribución, seleccione Crear ajustes de distribución. Esto abre la página Crear ajustes de distribución.
4. En la sección Tipo de imagen, elija Imagen de máquina de Amazon (AMI) como tipo de salida. Este es el valor predeterminado.
5. En la sección General, ingrese el nombre del recurso de ajustes de distribución que desea crear (obligatorio).

6. En la sección de configuración regional, seleccione el nombre de una plantilla de EC2 lanzamiento de la lista. Si no hay plantillas de lanzamiento en su cuenta, seleccione Crear nueva plantilla de lanzamiento, que abrirá las plantillas de lanzamiento en el EC2panel de control.

   Seleccione la casilla de verificación Establecer la versión predeterminada para actualizar la versión predeterminada de la plantilla de lanzamiento a la nueva versión que Generador de Imágenes crea con la AMI de salida.

   Para añadir otra plantilla de lanzamiento a la región seleccionada, seleccione Añadir configuración de plantilla de lanzamiento.

   Para eliminar una plantilla de lanzamiento, seleccione Eliminar.

7. Continúe especificando cualquier configuración adicional que necesite y seleccione Crear configuración para crear su nuevo recurso de ajustes de distribución.

Agregue una plantilla de EC2 lanzamiento a la configuración de distribución de AMI desde AWS CLI

En esta sección se describe cómo configurar un archivo de ajustes de distribución con una plantilla de lanzamiento y cómo utilizar el comando create-image en la AWS CLI para crear y distribuir una AMI de Generador de Imágenes y una nueva versión de la plantilla de lanzamiento que la utilice.

1. Configuración de un archivo de ajustes de distribución

   Antes de poder crear una AMI de Image Builder con una plantilla de lanzamiento AWS CLI, debe crear una estructura JSON de configuración de distribución que especifique los `launchTemplateConfigurations` ajustes. Tiene que especificar al menos una entrada `launchTemplateConfigurations` en la región de origen.

   El siguiente archivo de ejemplo, denominado `create-distribution-config-launch-template.json`, muestra algunos escenarios posibles para la configuración de la plantilla de lanzamiento en la región de origen.

```
{
    "name": "NewDistributionConfiguration",
    "description": "This is just a test",
    "distributions": [
        {
            "region": "us-west-2",
```

```
            "amiDistributionConfiguration": {
                "name": "test-{{imagebuilder:buildDate}}-
{{imagebuilder:buildVersion}}",
                "description": "description"
            },
            "launchTemplateConfigurations": [
                {
                    "launchTemplateId": "lt-0a1bcde2fgh34567",
                    "accountId": "935302948087",
                    "setDefaultVersion": true
                },
                {
                    "launchTemplateId": "lt-0aaa1bcde2ff3456"
                },
                {
                    "launchTemplateId": "lt-12345678901234567",
                    "accountId": "123456789012"
                }
            ]
        }
    ],
    "clientToken": "clientToken1"
}
```

2. Creación de los ajustes de distribución

   Para crear un recurso de configuración de distribución de Image Builder mediante el create-distribution-configurationcomando del AWS CLI, proporcione los siguientes parámetros en el comando:

   - Ingrese el nombre de la distribución en el parámetro `--name`.
   - Adjunte el archivo JSON de configuración de distribución que creó en el parámetro `--cli-input-json`.

```
aws imagebuilder create-distribution-configuration --name my distribution name--
cli-input-json file://create-distribution-config-launch-template.json
```

> **Note**
>
> - Debe incluir la notación `file://` al principio de la ruta del archivo JSON.

- La ruta del archivo JSON debe seguir la convención apropiada para el sistema operativo base donde se está ejecutando el comando. Por ejemplo, Windows utiliza la barra diagonal inversa (\) para referirse a la ruta de directorios, mientras que Linux y macOS usan la barra diagonal (/).

También puede proporcionar JSON directamente en el comando, mediante el parámetro `--distributions`.

Utilice capacidades de distribución de AMI mejoradas

Image Builder ofrece funciones de distribución avanzadas que le proporcionan flexibilidad y control sobre la forma en AMIs que se distribuyen entre regiones y cuentas. Estas capacidades separan la distribución del proceso de creación, lo que le permite distribuir las imágenes existentes a pedido, recuperarse de los errores de distribución de manera eficiente e implementar estrategias de distribución controladas y en varias etapas mediante flujos de trabajo personalizables.

Puede utilizar las funciones de distribución de AMI mejoradas de Image Builder para realizar directamente las actividades de distribución sin necesidad de volver a ejecutar una creación de imagen completa.

Distribución desacoplada

La DistributeImage API acepta tres tipos de referencias a imágenes de origen:

- ID de AMI: un identificador de AMI estándar (por ejemplo,`ami-0abcdef1234567890`)
- Parámetro SSM: parámetro SSM que almacena un ID de AMI (por ejemplo,) `ssm:/my/ami/parameter`
- Image Builder versión ARN: versión de imagen ARN de Image Builder

Reintento de distribución

Si se produce un error en la distribución de una imagen, utilice la `RetryImage` API para volver a intentarlo. Esto reduce el tiempo necesario para solucionar la causa del error al evitar una reconstrucción completa de la imagen. Úselo `RetryImage` después de resolver la causa subyacente del error de distribución.

La RetryImage API acepta un ARN de la versión de compilación de imágenes (por ejemplo,`arn:aws:imagebuilder:us-west-2:123456789012:image/my-image/1.0.0/1`). Al invocar la API, Image Builder reanuda automáticamente el proceso de distribución desde el punto del error utilizando la configuración y los ajustes de distribución originales. La `RetryImage` API puede volver a intentar las distribuciones que fallaron durante la fase de distribución, la fase de prueba o la fase de integración. Funciona AMIs en los siguientes estados: pendiente, fallido, eliminado o disponible.

Requisitos previos

Antes de volver a intentar una distribución, asegúrese de lo siguiente:

- Ha identificado y resuelto la causa raíz del error. Revise los registros de distribución en CloudWatch Logs para ver los detalles del error.
- Dispone de los permisos de IAM necesarios para volver a intentar crear la imagen.
- En caso de errores en la distribución entre cuentas, compruebe que `EC2ImageBuilderDistributionCrossAccountRole` la cuenta de destino tenga la `Ec2ImageBuilderCrossAccountDistributionAccess` política adjunta.

Importante: si vuelve a intentarlo sin solucionar el problema subyacente, se producirán errores repetidos.

flujos de trabajo de distribución

Los flujos de trabajo de distribución son un nuevo tipo de flujo de trabajo que complementa los flujos de trabajo de creación y prueba, lo que permite definir y controlar el proceso de distribución con pasos secuenciales. Con los flujos de trabajo de distribución, puede crear procesos de distribución personalizados que incluyan operaciones de copia de la AMI, wait-for-action puntos de control, modificaciones de atributos de imagen y otros pasos relacionados con la distribución. Esto proporciona un control estructurado sobre la forma en AMIs que se distribuye, con visibilidad escalonada, capacidades de distribución paralela e informes de errores detallados.

Para obtener más información sobre cómo crear y personalizar flujos de trabajo, consulte Administrar flujos de trabajo de imágenes.

Comparta los recursos de Image Builder con AWS RAM

EC2 Image Builder se integra con AWS Resource Access Manager (AWS RAM) para que puedas compartir los siguientes tipos de recursos de Image Builder con cualquiera Cuenta de AWS o a través de ellos AWS Organizations.

- Componentes
- Imágenes
- Recetas

Para compartir recursos a través de ellos AWS RAM, debe crear un recurso compartido. Un uso compartido de recursos especifica los recursos para compartir y los consumidores con quienes compartirlos. Los consumidores pueden ser individuos Cuentas de AWS, unidades organizativas o toda una organización AWS Organizations. La siguiente lista incluye los tipos de cuentas y organizaciones con las que puede compartir.

- Cuentas de AWS Específico dentro o fuera de su organización en AWS Organizations.
- Una unidad organizativa (OU) dentro de la organización en AWS Organizations.
- Toda la organización en AWS Organizations.
- AWS Organizations o OUs fuera de su organización.

En este modelo, el Cuenta de AWS propietario del recurso (propietario) lo comparte con otros Cuentas de AWS o a través de AWS Organizations (consumidores) dentro de la misma región. Cuando se actualiza un recurso compartido, los consumidores reciben esas actualizaciones automáticamente.

> **Note**
>
> Los componentes, imágenes y recetas de imágenes compartidas se tienen en cuenta únicamente para los límites de recursos correspondientes del propietario. Los límites de recursos de los consumidores no se ven afectados por los recursos que se comparten con ellos.

Temas

- Propietarios del recurso
- Consumidores de recursos
- Cree un AWS RAM recurso compartido para sus recursos de Image Builder
- Dejar de compartir un recurso de Image Builder de AWS RAM

Propietarios del recurso

Los recursos de Image Builder solo se pueden compartir en el Región de AWS lugar donde se crearon. Cuando compartes estos recursos, no se replicarán en todas las regiones.

Para obtener una lista de los recursos de Generador de imágenes de los que es propietario y que puede compartir, especifique el filtro de propiedad en la consola o cuando ejecute el comando en la AWS CLI.

- Enumeración de componentes de Generador de imágenes
- Enumeración de imágenes
- Enumeración y visualización de los detalles de una receta de imágenes
- Enumeración y visualización de los detalles de una receta de contenedor

Para obtener más información al respecto AWS RAM, consulte la Guía AWS RAM del usuario.

Requisitos previos para compartir recursos de Generador de imágenes

Para compartir un recurso de Generador de imágenes como un componente, una imagen o receta:

- Cuenta de AWS Debe ser propietario del recurso Image Builder que desee compartir. No puede compartir recursos que se han compartido con usted.
- La clave AWS Key Management Service (AWS KMS) asociada a los recursos cifrados debe compartirse de forma explícita con las cuentas, organizaciones o entidades de destino OUs.
- Para poder compartir sus recursos de Image Builder AWS Organizations y OUs usarlos AWS RAM, debe habilitar el uso compartido. Para obtener más información, consulte Habilitar el uso compartido con AWS Organizations en la Guía del usuario de AWS RAM .
- Si distribuye una imagen cifrada AWS KMS entre cuentas de distintas regiones, debe crear una clave de KMS y un alias en cada región de destino. Además, las personas que vayan a lanzar instancias en esas regiones deberán acceder a la clave de KMS especificada en la política de claves.

Los siguientes recursos que Image Builder crea a partir de su creación en proceso no se consideran recursos de Image Builder, sino que son recursos externos que Image Builder distribuye en su cuenta y a las Regiones de AWS cuentas y organizaciones o unidades organizativas (OUs) que especifique en la configuración de distribución.

- Imágenes de máquinas de Amazon (AMIs)
- Imágenes de contenedor que residen en Amazon ECR

Para obtener más información sobre los ajustes de distribución para su AMI, consulte Creación y actualización de ajustes de distribución de AMI. Para obtener más información sobre los ajustes de distribución de la imagen de contenedor en Amazon ECR, consulte Creación y actualización de los ajustes de la distribución de las imágenes de los contenedores.

Para obtener más información sobre cómo compartir su AMI con AWS Organizations y OUs, consulte Compartir una AMI con organizaciones o OUs.

Consumidores de recursos

Los consumidores pueden usar un recurso compartido, pero no pueden modificarlo de ninguna manera. Cuando crean recetas de Generador de imágenes, pueden especificar una imagen compartida como imagen base y pueden añadir componentes compartidos. También pueden especificar una receta compartida cuando crean una canalización de imágenes de Generador de imágenes o cuando utilizan el comando create-image en la AWS CLI.

Si pertenece a una organización y está habilitado el uso compartido dentro de su organización, los consumidores de su organización tienen acceso automático al recurso compartido. AWS Organizations De lo contrario, los consumidores reciben una invitación para unirse al recurso compartido y se les concede acceso al recurso compartido después de aceptar la invitación.

Cree un AWS RAM recurso compartido para sus recursos de Image Builder

Para compartir un componente, una imagen o una receta de Image Builder, debe agregarlo a un AWS Resource Access Manager recurso compartido. El uso compartido de recursos especifica los recursos que se deben compartir y los consumidores con quienes se comparten.

Dispone de las siguientes opciones para compartir sus recursos.

Opción 1: crear un recurso compartido de RAM

Al crear un recurso compartido de RAM, puede compartir un componente, imagen o receta de su propiedad en un solo paso. Utilice uno de los siguientes métodos para crear el recurso compartido:

- Consola

  Para crear un recurso compartido mediante la AWS RAM consola, consulte Compartir AWS los recursos de su propiedad en la Guía del AWS RAM usuario.

- AWS CLI

  Para crear el recurso compartido mediante la interfaz de línea de AWS RAM comandos, ejecute el create-resource-share comando en AWS CLI.

Opción 2: aplicar una política de recursos y convertirla en un recurso compartido existente

La segunda opción para compartir los recursos consta de dos pasos: ejecutar AWS CLI los comandos en ambos. El primer paso utiliza los comandos de Image Builder AWS CLI para aplicar políticas basadas en recursos al recurso compartido. El segundo paso convierte el recurso en un recurso compartido de RAM mediante el promote-resource-share-created-from-policy AWS RAM comando incluido en el AWS CLI para garantizar que el recurso esté visible para todas las personas con las que lo has compartido.

1. Aplicar la política de recursos

   Para aplicar correctamente la política de recursos, debe asegurarse de que la cuenta con la que comparte tiene permiso para acceder a los recursos subyacentes.

   Elija la pestaña que coincida con su tipo de recurso para el comando correspondiente.

   Image

   Puede aplicar una política de recursos a una imagen para permitir que otros la utilicen como imagen base en sus recetas.

   Ejecute el comando put-image-policy Image Builder en el AWS CLI, para identificar AWS los principales con los que compartir la imagen.

```
aws imagebuilder put-image-policy --image-arn arn:aws:imagebuilder:us-
west-2:123456789012:image/my-example-image/2019.12.03/1 --policy '{ "Version":
 "2012-10-17",        "Statement": [ { "Effect": "Allow", "Principal":
 { "AWS": [ "123456789012" ] }, "Action": ["imagebuilder:GetImage",
 "imagebuilder:ListImages"], "Resource": [ "arn:aws:imagebuilder:us-
west-2:123456789012:image/my-example-image/2019.12.03/1" ] } ] }'
```

Component

Puede aplicar una política de recursos a un componente de compilación o prueba para permitir el uso compartido entre cuentas. Este comando permite a otras cuentas usar su componente en sus recetas. Para aplicar correctamente la política de recursos, debe asegurarse de que la cuenta con la que comparte tiene permiso para acceder a los recursos a los que hace referencia el componente compartido, como los archivos alojados en repositorios privados.

Ejecute el comando put-component-policy Image Builder en AWS CLI, para identificar AWS los principales con los que compartir el componente.

```
aws imagebuilder put-component-policy --component-arn arn:aws:imagebuilder:us-
west-2:123456789012:component/my-example-component/2019.12.03/1 --
policy '{ "Version": "2012-10-17",        "Statement": [ { "Effect":
 "Allow", "Principal": { "AWS": [ "123456789012" ] }, "Action":
 [ "imagebuilder:GetComponent", "imagebuilder:ListComponents" ], "Resource":
 [ "arn:aws:imagebuilder:us-west-2:123456789012:component/my-example-
component/2019.12.03/1" ] } ] }'
```

Image recipe

Puede aplicar una política de recursos a una receta de imágenes para permitir el uso compartido entre cuentas. Este comando permite a otras cuentas usar su receta para crear imágenes en sus cuentas. Para aplicar correctamente la política de recursos, debe asegurarse de que la cuenta con la que comparte tiene permiso para acceder a los recursos a los que hace referencia la receta, como la imagen base o los componentes seleccionados.

Ejecute el comando put-image-recipe-policy Image Builder en el AWS CLI, para identificar AWS los principales con los que compartir la imagen.

```
aws imagebuilder put-image-recipe-policy --image-recipe-arn
 arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/my-example-image-
recipe/2019.12.03 --policy '{ "Version": "2012-10-17",        "Statement":
 [ { "Effect": "Allow", "Principal": { "AWS": [ "123456789012" ] }, "Action":
 [ "imagebuilder:GetImageRecipe", "imagebuilder:ListImageRecipes" ], "Resource":
 [ "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/my-example-image-
recipe/2019.12.03" ] } ] }'
```

Container recipe

Puede aplicar una política de recursos a una receta de contenedor para permitir el uso compartido entre cuentas. Este comando permite a otras cuentas usar su receta para crear imágenes en sus cuentas. Para aplicar correctamente la política de recursos, debe asegurarse de que la cuenta con la que comparte tiene permiso para acceder a los recursos a los que hace referencia la receta, como la imagen base o los componentes seleccionados.

Ejecute el comando put-container-recipe-policy Image Builder en el AWS CLI, para identificar AWS los principales con los que compartir la imagen.

```
aws imagebuilder put-container-recipe-policy --container-recipe-arn
 arn:aws:imagebuilder:us-west-2:123456789012:container-recipe/my-
example-container-recipe/2021.12.03 --policy '{ "Version": "2012-10-17",
        "Statement": [ { "Effect": "Allow", "Principal": { "AWS":
 [ "123456789012" ] }, "Action": [ "imagebuilder:GetContainerRecipe",
 "imagebuilder:ListContainerRecipes" ], "Resource": [ "arn:aws:imagebuilder:us-
west-2:123456789012:container-recipe/my-example-container-
recipe/2021.12.03" ] } ] }'
```

> **Note**
>
> Para establecer las políticas correctas para compartir y dejar de compartir un recurso, el propietario del recurso debe tener permisos `imagebuilder:put*`.

2. Cómo convertir en un recurso compartido de RAM

Para asegurarse de que el recurso esté visible para todos los responsables con los que lo ha compartido, ejecute el promote-resource-share-created-from-policy AWS RAM comando incluido en el. AWS CLI

Dejar de compartir un recurso de Image Builder de AWS RAM

Para dejar de compartir un recurso de Image Builder que sea de su propiedad, como un componente, una imagen o una receta compartidos, debe eliminarlo del AWS Resource Access Manager recurso compartido. Para ello, puede utilizar la consola de AWS RAM o la AWS CLI.

> **Note**
>
> Los propietarios no pueden eliminar un recurso compartido hasta que ya no se esté compartiendo. Un propietario no puede dejar de compartir estos recursos hasta que ninguno de los consumidores dependa de ellos.

Para dejar de compartir un componente, imagen o receta compartida que posea utilizando la consola de AWS Resource Access Manager

Consulte Actualizar un recurso compartido en la Guía del usuario de AWS RAM .

Para dejar de compartir un componente, imagen o receta compartida que posea utilizando AWS CLI

Use el comando disassociate-resource-share para dejar de compartir el recurso.

Etiquetado de los recursos de salida de Generador de imágenes

Etiquetar los recursos puede resultar útil para filtrar y realizar un seguimiento de los costos de los recursos o de otras categorías. También puede controlar el acceso mediante etiquetas. Para obtener más información acerca de la autorización basada en etiquetas, consulte Autorización basada en etiquetas de Generador de Imágenes

Generador de Imágenes es compatible con las siguientes etiquetas dinámicas:

- `- {{imagebuilder:buildDate}}`

  Resuelve construir date/time en el momento de la construcción.

- `- {{imagebuilder:buildVersion}}`

  Se resuelve en una versión de compilación, que es un número que se encuentra al final de un Nombre de recurso de Amazon (ARN) de Generador de imágenes. Por ejemplo, `"arn:aws:imagebuilder:us-west-2:123456789012:component/myexample-component/2019.12.02/1"` muestra la versión de compilación como 1.

Para ayudarle a realizar un seguimiento de las imágenes de Amazon Machine (AMIs) que ha distribuido, Image Builder añade automáticamente las siguientes etiquetas a la salida AMIs.

- `"CreatedBy":"EC2 Image Builder"`
- `"Ec2ImageBuilderArn":"arn:aws:imagebuilder:`*`us-west-2`*`:`*`123456789012`*`:image/`*`simple-recipe-linux`*`/`*`1.0.0/10`*`"`. Esta etiqueta contiene el ARN del recurso de imagen de Generador de imágenes que se utilizó para crear la AMI.

Contenido

- Etiquete un recurso del AWS CLI
- Quite la etiqueta de un recurso del AWS CLI
- Enumera todas las etiquetas de un recurso específico del AWS CLI

Etiquete un recurso del AWS CLI

El siguiente ejemplo muestra cómo utilizar un comando imagebuilder CLI para añadir y etiquetar un recurso en EC2 Image Builder. Debe proporcionar `resourceArn` y las etiquetas que desee aplicárselas.

Los contenidos del ejemplo `tag-resource.json` son los siguientes:

```
{
    "resourceArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/my-example-pipeline",
    "tags": {
        "KeyName": "KeyValue"
    }
}
```

Ejecute el siguiente comando que hace referencia al archivo `tag-resource.json` anterior.

```
aws imagebuilder tag-resource --cli-input-json file://tag-resource.json
```

Quite la etiqueta de un recurso del AWS CLI

En el ejemplo siguiente se indica cómo utilizar un comando de la CLI imagebuilder para eliminar una etiqueta de un recurso. Debe proporcionar `resourceArn` y las claves para eliminar la etiqueta.

Los contenidos del ejemplo `untag-resource.json` son los siguientes:

```
{
    "resourceArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/my-example-pipeline",
    "tagKeys": [
        "KeyName"
    ]
}
```

Ejecute el siguiente comando que hace referencia al archivo `untag-resource.json` anterior.

```
aws imagebuilder untag-resource --cli-input-json file://untag-resource.json
```

Enumera todas las etiquetas de un recurso específico del AWS CLI

El siguiente ejemplo muestra cómo utilizar un comando de la CLI imagebuilder para enumerar todas las etiquetas de un recurso específico.

```
aws imagebuilder list-tags-for-resource --resource-arn arn:aws:imagebuilder:us-
west-2:123456789012:image-pipeline/my-example-pipeline
```

Eliminación de recursos de Generador de imágenes obsoletos o no utilizados

Su entorno Generador de Imágenes, al igual que su hogar, necesita un mantenimiento habitual para ayudarlo a encontrar lo que necesita y completar sus tareas sin tener que preocuparse por el desorden. Asegúrese de limpiar periódicamente los recursos temporales que creó para las pruebas. De lo contrario, es posible que se olvide de esos recursos y, más adelante, no recuerde para qué se utilizaron. Para entonces, es posible que no esté claro si puede deshacerse de ellos de manera segura.

La eliminación de recursos no elimina ninguna imagen de contenedor de Amazon EC2 AMIs o Amazon ECR que se haya creado durante el proceso de creación de la imagen. Debe limpiarlos por separado, utilizando las acciones de la consola, la API EC2 o los AWS CLI comandos correspondientes de Amazon o Amazon ECR.

> **Tip**
>
> Para evitar errores de dependencia al eliminar recursos, asegúrese de eliminarlos en el siguiente orden:
>
> 1. Canalización de imágenes
> 2. Receta de imagen
> 3. Todos los recursos restantes

Elimine los recursos del Consola de administración de AWS

Para eliminar una canalización de imágenes y sus recursos, siga estos pasos:

Eliminación de la canalización

1. Para ver una lista de las canalizaciones de compilación creadas en su cuenta, seleccione las canalizaciones de imágenes en el panel de navegación.
2. Seleccione la casilla de verificación junto al nombre de canalización para seleccionar la canalización que desea eliminar.

3. En la parte superior del panel Canalizaciones de imágenes, en el menú Acciones, seleccione Eliminar.
4. Para confirmar la eliminación, ingrese `Delete` en el cuadro y seleccione Eliminar.

Eliminación de la receta

1. Para ver una lista de las recetas creadas en su cuenta, seleccione Recetas de imagen en el panel de navegación.
2. Seleccione la casilla de verificación junto al nombre de la receta para seleccionar la receta que desea eliminar.
3. En la parte superior del panel Recetas de imagen, en el menú Acciones, seleccione Eliminar receta.
4. Para confirmar la eliminación, ingrese `Delete` en el cuadro y seleccione Eliminar.

Eliminación de configuración de infraestructura

1. Para ver una lista de las configuraciones de infraestructura creadas en su cuenta, seleccione Configuración de infraestructura en el panel de navegación.
2. Seleccione la casilla de verificación junto al nombre de la configuración para seleccionar la configuración de infraestructura que desea eliminar.
3. En la parte superior del panel Configuraciones de infraestructura, seleccione Eliminar.
4. Para confirmar la eliminación, ingrese `Delete` en el cuadro y seleccione Eliminar.

Eliminación de ajustes de distribución

1. Para ver una lista de los ajustes de distribución creados en su cuenta, seleccione Ajustes de distribución en el panel de navegación.
2. Seleccione la casilla de verificación junto al nombre de la configuración para seleccionar los ajustes de distribución que creó para este tutorial.
3. En la parte superior del panel Ajustes de distribución, seleccione Eliminar.
4. Para confirmar la eliminación, ingrese `Delete` en el cuadro y seleccione Eliminar.

Eliminación de una imagen

1. Para ver una lista de las imágenes creadas en su cuenta, seleccione Imágenes en el panel de navegación.
2. Elija la versión de la imagen que desea eliminar. Esto abre la página Versiones de compilación de imágenes.
3. Seleccione la casilla de verificación junto a la versión de cualquier imagen que desea eliminar. Puede seleccionar más de una versión de imagen a la vez.
4. En la parte superior del panel Versiones de compilación de imágenes, seleccione Eliminar versión.
5. Para confirmar la eliminación, ingrese `Delete` en el cuadro y seleccione Eliminar.

Elimine una canalización de imágenes del AWS CLI

Los siguientes ejemplos muestran cómo eliminar los recursos de Generador de Imágenes utilizando la AWS CLI. Como se mencionó anteriormente, los recursos se deben eliminar en el siguiente orden para evitar errores de dependencia:

1. Canalización de imágenes
2. Receta de imagen
3. Todos los recursos restantes

Elimine la canalización de imágenes del AWS CLI

En el siguiente ejemplo se muestra cómo eliminar una canalización de imágenes al especificar su ARN.

```
aws imagebuilder delete-image-pipeline --image-pipeline-arn arn:aws:imagebuilder:us-
west-2:123456789012:image-pipeline/my-example-pipeline
```

Elimine la receta de imagen del AWS CLI

En el siguiente ejemplo se muestra cómo eliminar una receta de imagen al especificar su ARN.

```
aws imagebuilder delete-image-recipe --image-recipe-arn arn:aws:imagebuilder:us-
west-2:123456789012:image-recipe/my-example-recipe/2019.12.03
```

Eliminación de una configuración de infraestructura

En el siguiente ejemplo se muestra cómo eliminar un recurso de configuración de infraestructura al especificar su ARN.

```
aws imagebuilder delete-infrastructure-configuration --infrastructure-configuration-arn
 arn:aws:imagebuilder:us-west-2:123456789012:infrastructure-configuration/my-example-
infrastructure-configuration
```

Eliminación de ajustes de distribución

En el siguiente ejemplo se muestra cómo eliminar un recurso de ajustes de distribución al especificar su ARN.

```
aws imagebuilder delete-distribution-configuration --distribution-configuration-arn
 arn:aws:imagebuilder:us-west-2:123456789012:distribution-configuration/my-example-
distribution-configuration
```

Eliminación de una imagen

En el siguiente ejemplo se muestra cómo eliminar una versión de compilación de imágenes al especificar su ARN.

```
aws imagebuilder delete-image --image-build-version-arn arn:aws:imagebuilder:us-
west-2:123456789012:image/my-example-image/2019.12.02/1
```

Eliminación de un componente

En el siguiente ejemplo se muestra cómo usar un comando imagebuilder de la CLI para eliminar una versión de compilación de componentes al especificar su ARN.

```
aws imagebuilder delete-component --component-build-version-arn
 arn:aws:imagebuilder:us-west-2:123456789012:component/my-example-
component/2019.12.02/1
```

> **Important**
>
> Asegúrese de que no haya recetas que hagan referencia a la versión de compilación de componentes de ninguna manera antes de eliminarla. De lo contrario, se podrían producir fallos en la canalización.

Administración de flujos de trabajo de creación y prueba para imágenes de Generador de imágenes

Un flujo de trabajo de imágenes define la secuencia de pasos que EC2 Image Builder lleva a cabo durante las etapas de creación, prueba y distribución del proceso de creación de imágenes. Forma parte del marco general de flujos de trabajo de Generador de imágenes.

Ventajas de los flujos de trabajo de imágenes

- Con los flujos de trabajo de imágenes, tiene más flexibilidad, visibilidad y control sobre el proceso de creación de imágenes.
- Puede agregar pasos de flujo de trabajo personalizados al definir el documento de flujos de trabajo o puede optar por utilizar el flujo de trabajo predeterminado de Generador de imágenes.
- Puede excluir los pasos del flujo de trabajo que se incluyen en los flujos de trabajo de imágenes predeterminados.
- Puede crear flujos de trabajo solo de prueba que omitan por completo el proceso de creación. Puede hacer lo mismo para crear flujos de trabajo exclusivos de compilación o distribución.

> **Note**
>
> No puede modificar un flujo de trabajo existente, pero puede clonarlo o crear una nueva versión.

Temas de flujo de trabajo

- Marco del flujo de trabajo: etapas
- Acceso a los servicios
- Utilice flujos de trabajo gestionados para sus imágenes
- Enumeración de flujos de trabajo de imágenes
- Creación de un flujo de trabajo de imágenes
- Creación de un documento de flujos de trabajo YAML

Marco del flujo de trabajo: etapas

Para personalizar los flujos de trabajo de imágenes, es importante comprender las etapas del flujo de trabajo que componen el marco del flujo de trabajo de creación de imágenes.

El marco del flujo de trabajo de creación de imágenes incluye las siguientes etapas distintas.

1. Etapa de compilación (previa a la instantánea): durante la etapa de compilación, realizas cambios en la instancia de EC2 compilación de Amazon en la que se ejecuta tu imagen base para crear la línea base de la nueva imagen. Por ejemplo, la receta puede incluir componentes que instalen una aplicación o modifiquen la configuración del firewall del sistema operativo.

   Una vez que esta etapa se complete correctamente, Generador de Imágenes crea una instantánea o imagen de contenedor que utilizará para la etapa de prueba y posteriores.
2. Etapa de prueba (posterior a la instantánea): durante la etapa de prueba, existen algunas diferencias entre las imágenes que crean imágenes AMIs y las que contienen imágenes. Para los flujos de trabajo de AMI, Image Builder lanza una EC2 instancia a partir de la instantánea que creó como paso final de la fase de creación. Las pruebas se ejecutan en la nueva instancia para validar la configuración y garantizar que la instancia funcione según lo previsto. En el caso de los flujos de trabajo de contenedor, las pruebas se ejecutan en la misma instancia que se utilizó para la creación.
3. Etapa de distribución (posterior a la creación): durante la etapa de distribución, una vez que se ha completado la creación de la imagen y se han superado satisfactoriamente todas las pruebas, la imagen de salida pasa por varias fases de distribución. Estas fases abarcan las operaciones de copia de la AMI, las modificaciones de los atributos de las imágenes, el intercambio de imágenes y otras etapas de distribución relacionadas.

Acceso a los servicios

Para ejecutar flujos de trabajo de imágenes, Generador de imágenes necesita permiso para realizar acciones de flujo de trabajo. Puede especificar el rol vinculado al servicio AWSServiceRoleForImageBuilder, o bien puede especificar su propio rol personalizado para el acceso al servicio, de la siguiente manera.

- Consola: en el paso 3 del asistente de canalización Definir el proceso de creación de imágenes, seleccione el rol vinculado al servicio o su propio rol personalizado de la lista Rol de IAM del panel Acceso al servicio.

- API Image Builder: en la solicitud de CreateImageacción, especifique el rol vinculado al servicio o su propio rol personalizado como valor del parámetro. `executionRole`

Para obtener más información sobre cómo crear un rol de servicio, consulte Crear un rol para delegar permisos a un AWS servicio en la Guía del AWS Identity and Access Management usuario.

Utilice flujos de trabajo gestionados para sus imágenes

Los flujos de trabajo gestionados los crea y mantiene AWS. Cuando utiliza flujos de trabajo gestionados en sus canalizaciones de imágenes o para la creación de imágenes únicas, puede seleccionar el nombre de recurso de Amazon (ARN) del flujo de trabajo gestionado que desee utilizar. Amazon proporciona las versiones más recientes a las que se han aplicado parches y otras actualizaciones. Para obtener una lista de los flujos de trabajo gestionadosEnumeración de flujos de trabajo de imágenes, consulta y filtra por Owner = Amazon (consola).

Enumeración de flujos de trabajo de imágenes

En la página de lista Flujos de trabajo de imágenes de la consola de Generador de imágenes, puede ver una lista de los recursos de flujos de trabajo de imágenes de los que dispone o a los que tiene acceso, junto con algunos detalles clave sobre esos recursos. También puedes usar comandos o acciones con la API Image Builder o AWS CLI enumerar los flujos de trabajo de imágenes en tu cuenta. SDKs

Puede usar uno de los siguientes métodos para enumerar los recursos de flujos de trabajo de imágenes de los que dispone o a los que tiene acceso. Para ver la acción de la API, consulte ListWorkflowsla Referencia de la API de EC2 Image Builder. Para la solicitud de SDK asociada, consulte el enlace Véase también en la misma página.

Puede filtrar los siguientes detalles para simplificar la lista de resultados. Por ejemplo, si filtra Owner = Amazon en la consola, la lista solo muestra los flujos de trabajo AWS gestionados.

- Flujo de trabajo
- Versión
- Tipo
- Propietario

Console

Detalles del flujo de trabajo

Los detalles de la página de lista Flujos de trabajo de imágenes de la consola de Generador de imágenes incluyen lo siguiente:

- Flujo de trabajo: nombre de la versión más reciente del recurso de flujo de trabajo de imágenes. En la consola de Generador de imágenes, la columna Flujo de trabajo enlaza con la página de detalles del flujo de trabajo.
- Versión: versión más reciente del recurso de flujo de trabajo de imágenes.
- Tipo: el tipo de flujo de trabajo: `BUILDTEST`,, o`DISTRIBUTION`.
- Propietario: propietario del recurso de flujo de trabajo.
- Hora de creación: fecha y hora en que Generador de imágenes creó la versión más reciente del recurso de flujo de trabajo de imágenes.
- ARN: nombre de recurso de Amazon (ARN) de la versión actual del recurso de flujo de trabajo de imágenes.

Enumeración de flujos de trabajo de imágenes

Para enumerar los recursos de flujos de trabajo de imágenes en la consola de Generador de imágenes, lleve a cabo los siguientes pasos:

1. Abra la consola EC2 de Image Builder en https://console.aws.amazon.com/imagebuilder/.
2. Elija Flujos de trabajo de imágenes en el panel de navegación.

Cómo filtrar resultados

En la página de lista Flujos de trabajo de imágenes, puede buscar flujos de trabajo de imágenes específicos para filtrar los resultados. Los siguientes filtros están disponibles para los flujos de trabajo de imágenes:

`Workflow`

Para agilizar los resultados, puede ingresar la totalidad o parte del nombre del flujo de trabajo. La opción predeterminada es mostrar todos los flujos de trabajo en la lista.

`Version`

Para agilizar los resultados, puede ingresar la totalidad o parte del número de versión. La opción predeterminada es mostrar todas las versiones en la lista.

`Type`

Puede filtrar por tipo de flujo de trabajo o ver todos los tipos. La opción predeterminada es mostrar todos los tipos de flujo de trabajo en la lista.

- BUILD
- TEST
- DISTRIBUCIÓN

`Owner`

Al seleccionar el filtro de propietarios en la barra de búsqueda, Generador de imágenes muestra una lista de los propietarios de los flujos de trabajo de imágenes de su cuenta. Para agilizar los resultados, puede seleccionar un propietario de la lista. La opción predeterminada es mostrar todos los propietarios en la lista.

- Cuenta de AWS: cuenta que posee el recurso de flujo de trabajo.
- Amazon: recursos de flujo de trabajo que Amazon posee y administra.

AWS CLI

Al ejecutar el list-workflows comando en el AWS CLI, puede obtener una lista de los flujos de trabajo de imágenes de su propiedad o a los que tiene acceso.

En el siguiente ejemplo de comando se muestra cómo utilizar el comando list-workflows sin filtros y enumerar todos los recursos de flujos de trabajo de imágenes de Generador de imágenes que posee o a los que tiene acceso.

Ejemplo: enumeración de todos los flujos de trabajo de imágenes

```
aws imagebuilder list-workflows
```

Salida:

```
{
    "workflowVersionList": [
```

```
        {
            "name": "example-test-workflow",
            "dateCreated": "2023-11-21T22:53:14.347Z",
            "version": "1.0.0",
            "owner": "111122223333",
            "type": "TEST",
            "arn": "arn:aws:imagebuilder:us-west-2:111122223333:workflow/
test/example-test-workflow/1.0.0"
        },
        {
            "name": "example-build-workflow",
            "dateCreated": "2023-11-20T12:26:10.425Z",
            "version": "1.0.0",
            "owner": "111122223333",
            "type": "BUILD",
            "arn": "arn:aws:imagebuilder:us-west-2:111122223333:workflow/
build/example-build-workflow/1.0.0"
        },
        {
            "name": "example-distribution-workflow",
            "dateCreated": "2025-11-19T10:25:10.425Z",
            "version": "1.0.0",
            "owner": "111122223333",
            "type": "DISTRIBUTION",
            "arn": "arn:aws:imagebuilder:us-west-2:111122223333:workflow/
distribution/example-distribution-workflow/1.0.0"
        }
    ]
}
```

Al ejecutar el comando list-workflows, puede aplicar filtros para agilizar los resultados, como se muestra en el siguiente ejemplo. Para obtener más información sobre cómo filtrar los resultados, consulte el comando list-workflows en la Referencia de comandos de la AWS CLI .

Ejemplo: filtro para flujos de trabajo de creación

```
aws imagebuilder list-workflows --filters name="type",values="BUILD"
```

Salida:

```
{
    "workflowVersionList": [
```

```
        {
            "name": "example-build-workflow",
            "dateCreated": "2023-11-20T12:26:10.425Z",
            "version": "1.0.0",
            "owner": "111122223333",
            "type": "BUILD",
            "arn": "arn:aws:imagebuilder:us-west-2:111122223333:workflow/
build/example-build-workflow/1.0.0"
        }
    ]
}
```

Creación de un flujo de trabajo de imágenes

Cuando crea un flujo de trabajo de imágenes, tiene más control sobre el proceso de creación de imágenes. Puede especificar qué flujo de trabajo se ejecuta cuando Generador de imágenes crea la imagen y qué flujos de trabajo se ejecutan cuando prueba la imagen. También puede especificar una clave administrada por el cliente para cifrar los recursos del flujo de trabajo. Para obtener más información sobre el cifrado de los recursos del flujo de trabajo, consulte Cifrado y administración de claves en Generador de imágenes.

Para la creación de imágenes, puede especificar un flujo de trabajo en la etapa de creación y uno o varios flujos de trabajo en la etapa de prueba. Incluso puede omitir por completo la etapa de creación o prueba, en función de sus necesidades. Las acciones que realiza el flujo de trabajo se configuran en el documento de definición YAML que utiliza el flujo de trabajo. Para obtener más información sobre la sintaxis de los documentos YAML, consulte Creación de un documento de flujos de trabajo YAML.

Para conocer los pasos de creación de un nuevo flujo de trabajo de creación o prueba, seleccione la pestaña que coincida con el entorno que utilizará.

Consola de administración de AWS

Puede seguir el proceso a continuación para crear un nuevo flujo de trabajo en la consola de Generador de imágenes.

1. Abra la consola EC2 de Image Builder en https://console.aws.amazon.com/imagebuilder/.
2. Elija Flujos de trabajo de imágenes en el panel de navegación. Se mostrará una lista de los flujos de trabajo de imágenes que su cuenta posee o a los que tiene acceso.

> **Note**
>
> En la lista, siempre verá los recursos de flujo de trabajo administrados por Amazon que Generador de imágenes utiliza para los flujos de trabajo predeterminados. Para ver los detalles de esos flujos de trabajo, puede seleccionar el enlace Flujo de trabajo.

3. Para crear un nuevo flujo de trabajo, elija Crear flujo de trabajo de imágenes. Aparece la página Crear flujo de trabajo de imágenes.
4. Configure los detalles del nuevo flujo de trabajo. Para crear un flujo de trabajo de creación, seleccione la opción Creación situada en la parte superior del formulario. Para crear un flujo de trabajo de prueba, seleccione la opción Prueba situada en la parte superior del formulario. Generador de imágenes rellena la lista Plantillas en función de esta opción. Todos los demás pasos son los mismos para los flujos de trabajo de creación y prueba.

   General

   En la sección general se incluyen opciones que se aplican al recurso de flujo de trabajo, como el nombre y la descripción. La configuración general incluye lo siguiente:

   - Nombre del flujo de trabajo de imágenes (obligatorio): nombre del flujo de trabajo de imágenes. El nombre debe ser exclusivo en su cuenta. El nombre puede tener una longitud máxima de 128 caracteres. Los caracteres válidos incluyen letras, números, espacios, - y _.
   - Versión (obligatorio): versión semántica del recurso de flujo de trabajo que se va a crear (principal.secundaria.revisión).
   - Descripción (opcional): si lo desea, agregue una descripción del flujo de trabajo.
   - Clave de KMS (opcional): puede cifrar los recursos del flujo de trabajo con una clave administrada por el cliente. Para obtener más información, consulte Cifrado de flujos de trabajo de imágenes con una clave administrada por el cliente.

   Documento de definición

   El documento de flujos de trabajo YAML contiene toda la configuración del flujo de trabajo.

Introducción

- Para empezar con una plantilla predeterminada de Generador de imágenes como referencia para el flujo de trabajo, seleccione la opción Comenzar desde plantillas. Esta opción está seleccionada de forma predeterminada. Tras elegir qué plantilla utilizar de la lista Plantillas, se copia la configuración predeterminada de la plantilla que ha seleccionado en el Contenido del nuevo documento de flujos de trabajo, donde puede realizar cambios.
- Para definir el documento de flujos de trabajo desde cero, seleccione la opción Empezar desde cero. El Contenido se rellena con un breve resumen de algunas partes importantes del formato del documento para ayudarlo a empezar.

En el panel Contenido se incluye una barra de estado en la parte inferior en la que se muestran las advertencias o los errores del documento YAML. Para obtener más información sobre cómo crear un documento de flujos de trabajo YAML, consulte Creación de un documento de flujos de trabajo YAML.

5. Cuando haya completado el flujo de trabajo, o si quiere guardar el progreso y volver a él más adelante, seleccione Crear flujo de trabajo.

AWS CLI

Antes de ejecutar el create-workflow comando en AWS CLI, debe crear el documento YAML que contenga toda la configuración de su flujo de trabajo. Para obtener más información, consulte Creación de un documento de flujos de trabajo YAML.

En el siguiente ejemplo se muestra cómo crear un flujo de trabajo de creación con el comando create-workflow de la AWS CLI . El parámetro `--data` hace referencia a un documento YAML que contiene la configuración de creación del flujo de trabajo que cree.

Ejemplo: creación de un flujo de trabajo

```
aws imagebuilder create-workflow --name example-build-workflow --semantic-
version 1.0.0 --type BUILD --data file://example-build-workflow.yml
```

Salida:

```
{
```

```
 "workflowBuildVersionArn": "arn:aws:imagebuilder:us-west-2:111122223333:workflow/
build/example-build-workflow/1.0.0/1",
 "clientToken": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222"
}
```

En el siguiente ejemplo se muestra cómo crear un flujo de trabajo de prueba con el comando create-workflow de la AWS CLI . El parámetro `--data` hace referencia a un documento YAML que contiene la configuración de creación del flujo de trabajo que cree.

Ejemplo: creación de un flujo de trabajo de prueba

```
aws imagebuilder create-workflow --name example-test-workflow --semantic-
version 1.0.0 --type TEST --data file://example-test-workflow.yml
```

Salida:

```
{
 "workflowBuildVersionArn": "arn:aws:imagebuilder:us-west-2:111122223333:workflow/
test/example-test-workflow/1.0.0/1",
 "clientToken": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222"
}
```

Creación de un documento de flujos de trabajo YAML

El documento de definición del formato YAML configura los pasos de entrada, salida y flujo de trabajo para las etapas de creación y prueba del proceso de creación de la imagen. Puede empezar con plantillas que incluyan pasos estandarizados, o bien puede empezar desde cero para definir su propio flujo de trabajo. Tanto si utiliza una plantilla como si empieza desde cero, puede personalizar el flujo de trabajo para adaptarlo a sus necesidades.

Estructura de un documento de flujos de trabajo YAML

El documento de flujos de trabajo YAML que Generador de imágenes utiliza para realizar acciones de creación y prueba de imágenes se estructura de la siguiente manera.

- Identificación de documentos de flujo de trabajo
- Parámetros de entrada de documentos de flujo
- Flujo de documentos

- Resultados de documentos de flujo de trabajo

Identificación de documentos de flujo de trabajo

Identifica de forma única el flujo de trabajo. En esta sección se pueden incluir los siguientes atributos.

Campo	Description (Descripción)	Tipo	Obligatorio/a
Nombre	Nombre del documento de flujos de trabajo.	Cadena	No
description	Descripción del documento.	Cadena	No
schemaVersion	Versión del esquema del documento, actualmente 1.0.	Cadena	Sí

Ejemplo

```
---
name: sample-test-image
description: Workflow for a sample image, with extra configuration options exposed
 through workflow parameters.
schemaVersion: 1.0
```

Parámetros de entrada de documentos de flujo

En esta parte del documento de flujos de trabajo se definen los parámetros de entrada que la persona que llama puede especificar. Si no dispone de ningún parámetro, puede omitir esta sección. Si especifica parámetros, cada parámetro puede incluir los siguientes atributos.

Campo	Description (Descripción)	Tipo	Obligatorio/a	Restricciones
name	El nombre del parámetro.	Cadena	Sí	
description	Descripción del parámetro.	Cadena	No	
predeterminado	Si no se proporciona ningún valor, el valor predeterminado del parámetro. Si no incluye un valor predeterminado en la definición del parámetro, el valor del parámetro es obligatorio en tiempo de ejecución.	Coincide con el tipo de datos del parámetro.	No	
type	El tipo de datos del parámetro. Si no incluye el tipo de datos en la definición del parámetro, el tipo del parámetro toma como predeterm	Cadena	Sí	El tipo de datos del parámetro debe ser uno de los siguientes: • `string` • `integer`

Campo	Description (Descripción)	Tipo	Obligatorio/a	Restricciones
	inado un valor de cadena obligator io en tiempo de ejecución.			• `boolean` • `stringList`

Ejemplo

Especifique el parámetro en el documento de flujos de trabajo.

```
parameters:
  - name: waitForActionAtEnd
    type: boolean
    default: true
    description: "Wait for an external action at the end of the workflow"
```

Utilice el valor del parámetro en el documento de flujos de trabajo.

```
$.parameters.waitForActionAtEnd
```

Flujo de documentos

Especifica hasta 15 acciones de paso para el flujo de trabajo. Los pasos se ejecutan en el orden en que se definen en el documento de flujos de trabajo. En caso de error, se ejecuta una restauración en orden inverso, empezando por el paso en el que se produjo el error y siguiendo por los pasos anteriores.

Cada paso puede hacer referencia a la salida de cualquier acción de paso anterior. Esto se conoce como encadenamiento o referencia. Para hacer referencia al resultado de una acción de un paso anterior, puedes usar un JSONPath selector. Por ejemplo:

```
$.stepOutputs.step-name.output-name
```

Para obtener más información, consulte Uso de variables dinámicas en el documento de flujos de trabajo.

> **Note**
>
> Aunque el paso en sí no tenga un atributo de salida, todas las salidas de una acción de paso se incluyen en `stepOutput` para el paso.

Cada paso puede incluir los siguientes atributos.

Campo	Description (Descripción)	Tipo	Obligatorio/a	Valor predeterminado	Restricciones
acción	Acción de flujo de trabajo que realiza este paso.	Cadena	Sí		Debe ser una acción de paso admitida para los documentos de flujos de trabajo de Generador de imágenes.
`if`, seguido de un conjunto de instrucciones condicionales que modifican el operador `if`.	Las instrucciones condicionales agregan puntos de decisión sobre el flujo de control al cuerpo de los pasos del flujo de trabajo.	Dict	No		Generador de imágenes admite las siguientes instrucciones condicionales como modificadores del operador `if`: •

Campo	Description (Descripción)	Tipo	Obligatorio/a	Valor predeterminado	Restricciones
					Condiciones y modificadores de ramificación: `if`, `and`, `or`, `not`. Las condiciones de ramificación se especifican en una línea por sí mismas. • Operadores de comparación: `booleanEquals`, `numberEquals`, `numberGreaterThan`, `numberGreaterThanEquals`, `numberLes`

Campo	Description (Descripción)	Tipo	Obligatorio/a	Valor predeterminado	Restricciones
					`sThan` , `numberLessThanEquals` , `stringEquals` .
description	Descripción del paso.	Cadena	No		No se permiten cadenas vacías. Si se incluye, la longitud debe ser de 1 a 1024 caracteres.

Campo	Description (Descripción)	Tipo	Obligatorio/a	Valor predeterminado	Restricciones
inputs	Contiene los parámetros que debe ejecutar la acción del paso. Puede especificar los valores clave como valores estáticos o con una JSONPath variable que se resuelva en el tipo de datos correcto.	Dict	Sí		
name	El nombre del paso. Este nombre debe ser único dentro del documento de flujos de trabajo.	Cadena	Sí		Debe tener entre 3 y 128 caracteres de longitud. Puede incluir caracteres alfanuméricos y _. Sin espacios.

Campo	Description (Descripción)	Tipo	Obligatorio/a	Valor predeterminado	Restricciones
onFailure	Configura la acción que se debe realizar si el paso produce un error, de la siguiente manera. Comportamiento • `Abort`: genera un error en el paso, genera un error en el flujo de trabajo y no ejecuta ningún paso restante después del paso con error. Si la restauración está habilitad	Cadena	No	`Abort`	`Abort` | `Continue`

Campo	Description (Descripción)	Tipo	Obligatorio/a	Valor predeterminado	Restricciones
	a, esta comienza con el paso con error y continúa hasta que se restauren todos los pasos que la permiten. • `Continue`: genera un error en el paso, pero se siguen ejecutando los pasos restantes después del paso con error. En este caso, no hay ninguna restauración.				

Campo	Description (Descripción)	Tipo	Obligatorio/a	Valor predeterminado	Restricciones
rollbackEnabled	Configura si el paso se restaurará en caso de que se produzca un error. Puede usar un valor booleano estático o una JSONPath variable dinámica que se resuelva en un valor booleano.	Booleano	No	`true`	`true` | `false` | o una JSONPath variable que se resuelva como verdadera o falsa.
timeoutSeconds	Tiempo máximo, en segundos, que se ejecuta el paso antes de producir un error y volver a intentarlo, si se aplican los reintentos.	Entero	No	Depende del valor predeterminado definido para la acción del paso, si corresponde.	Entre 1 y 86 400 segundos (24 horas como máximo)

Ejemplo

```
steps:
  - name: LaunchTestInstance
    action: LaunchInstance
    onFailure: Abort
    inputs:
      waitFor: "ssmAgent"

  - name: ApplyTestComponents
    action: ExecuteComponents
    onFailure: Abort
    inputs:
      instanceId.$: "$.stepOutputs.LlaunchTestInstance.instanceId"

  - name: TerminateTestInstance
    action: TerminateInstance
    onFailure: Continue
    inputs:
      instanceId.$: "$.stepOutputs.LaunchTestInstance.instanceId"

  - name: WaitForActionAtEnd
    action: WaitForAction
    if:
      booleanEquals: true
      value: "$.parameters.waitForActionAtEnd"
```

Resultados de documentos de flujo de trabajo

Define las salidas del flujo de trabajo. Cada salida es un par de clave-valor que especifica el nombre de la salida y el valor. Puede utilizar las salidas para exportar datos en tiempo de ejecución que puedan utilizar los flujos de trabajo posteriores. Esta sección es opcional.

Cada salida que defina incluye los siguientes atributos.

Campo	Description (Descripción)	Tipo	Obligatorio/a
Nombre	El nombre de la salida. El nombre	Cadena	Sí

Campo	Description (Descripción)	Tipo	Obligatorio/a
	debe ser único en todos los flujos de trabajo que incluya en la canalización.		
valor	Valor de la salida. El valor de la cadena puede ser una variable dinámica, como un archivo de salida de una acción de paso. Para obtener más información, consulte Uso de variables dinámicas en el documento de flujos de trabajo.	Cadena	Sí

Ejemplo

Cree un ID de imagen de salida para el documento de flujos de trabajo con la salida del paso `createProdImage`.

```
outputs:
  - name: 'outputImageId'
    value: '$.stepOutputs.createProdImage.imageId'
```

Consulte la salida del flujo de trabajo en el siguiente flujo de trabajo.

```
$.workflowOutputs.outputImageId
```

Acciones de paso admitidas para el documento de flujos de trabajo

En esta sección se incluyen detalles de las acciones de paso que admite Generador de imágenes.

Términos que se usan en esta sección

AMI

Imagen de máquina de Amazon

ARN

Nombre de recurso de Amazon

Acciones admitidas

- ApplyImageConfigurations
- BootstrapInstanceForContainer
- CollectImageMetadata
- CollectImageScanFindings
- CreateImage
- DistributeImage
- ExecuteComponents
- ExecuteStateMachine
- LaunchInstance
- ModifyImageAttributes
- RegisterImage
- RunCommand
- RunSysPrep
- SanitizeInstance
- TerminateInstance
- WaitForAction
- WaitForSSMAgent

ApplyImageConfigurations

Este paso aplica varias configuraciones e integraciones a las distribuidas AMIs, como las configuraciones de licencias, las configuraciones de plantillas de lanzamiento, las configuraciones de exportación de S3, las configuraciones de EC2 Fast Launch y las configuraciones de parámetros de

Systems Manager. Las configuraciones se aplican a las imágenes distribuidas solo en la cuenta de origen, excepto a las configuraciones de parámetros de SSM, que se pueden aplicar entre cuentas.

Tiempo de espera predeterminado: 360 minutos

Tiempo de espera máximo: 720 minutos

Restauración: no hay ninguna restauración para esta acción de paso.

Entradas: en la siguiente tabla se incluyen las entradas admitidas para esta acción de paso.

Nombre de la entrada	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado/a	Restricciones
region	La región de la imagen.	Cadena	Sí		
licenseConfigurationArns	El ARN de configuración de licencia de la imagen.	Matriz	No		
launchTemplateConfigurations		Matriz	No		
launchTemplateConfigurations:launchTemplateId	El ID de la plantilla de lanzamiento que se va a aplicar a la imagen.	Cadena	Sí, si `launchTemplateConfigurations` está especificado		
launchTemplateConfigurations: ID de cuenta	La cuenta de la plantilla de lanzamiento que se va IDs a aplicar a la imagen.	Cadena	No		

Nombre de la entrada	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado/a	Restricciones
launchTemplateConfigurations:setDefaultVersion	La configuración de versión predeterminada de la plantilla de lanzamiento para la imagen.	Booleano	No		
s3 ExportConfiguration		Matriz	No		
S3ExportConfiguration: Nombre del rol	El nombre del rol de configuración de exportación de S3 para la imagen.	Cadena	Sí, si `s3ExportConfiguration` está especificado		
s3ExportConfiguration: diskImageFormat	El formato de imagen de disco de configuración de exportación S3 para la imagen.	Cadena	Sí, si `s3ExportConfiguration` se especifica		Valores permitidos: VMDK|RAW|VHD

Nombre de la entrada	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado/a	Restricciones
S3ExportConfiguration: Cubeta S3	El nombre del depósito de configuración de exportación de S3 de la imagen.	Cadena	Sí, si `s3ExportConfiguration` está especificado		
S3: prefijo S3 ExportConfiguration	El prefijo del depósito de configuración de exportación de S3 para la imagen.	Cadena	No		
fastLaunchConfigurations	La configuración de EC2 Fast Launch de la imagen.	Matriz	No		
fastLaunchConfigurationsLa configuración de inicio rápido de la imagen. ----sep----:enabled	EC2 Inicio rápido enabled/disabled de la imagen.	Booleano	Sí, si `fastLaunchConfigurations` se especifica		

Nombre de la entrada	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado/a	Restricciones
fastLaunchConfigurationsSí, si se especifica ----SEP----:SnapshotConfiguration	EC2 Inicio enabled/ disabled rápido de la imagen.	Asignación	No		
fastLaunchConfigurationsInicio rápido de la imagen. ----SEP----:SnapshotConfiguration: targetResourceCount	EC2 El recuento de recursos objetivo de Fast Launch para la imagen.	Entero	No		
fastLaunchConfigurations:maxParallelLaunches	EC2 Fast Launch: máximo de lanzamientos paralelos de la imagen.	Entero	No		

Nombre de la entrada	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado/a	Restricciones
fastLaunchConfigurationsLanzamiento rápido: máximo de lanzamientos paralelos para la imagen. ----SEP----:Launch Template			No		
fastLaunchConfigurations:LaunchTemplate ----SEP----:LaunchTemplate: launchTemplateId	EC2 ID de plantilla de lanzamiento de Fast Launch para la imagen.	Cadena	No		

Nombre de la entrada	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado/a	Restricciones
fastLaunchConfigurationsID de plantilla de lanzamiento rápido para la imagen. ----SEP----:Launch Template: launchTemplateName	EC2 Nombre de la plantilla de lanzamiento de Fast Launch para la imagen.	Cadena	No		
fastLaunchConfigurationsNombre de la plantilla de lanzamiento de Fast Launch para la imagen. ----SEP----:Launch Template: launchTemplateVersion	EC2 Versión de la plantilla de lanzamiento rápido para la imagen.	Cadena	No		
ssmParameterConfigurations	La configuración del parámetro SSM para la imagen.	Asignación	No		

Nombre de la entrada	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado/a	Restricciones
ssmParameterConfigurations: amiAccountId	El ID de la cuenta AMI del parámetro SSM de la imagen.	Cadena	No		
ssmParameterConfigurationsEl ID de la cuenta AMI del parámetro SSM para la imagen. ----SEP----:ParameterName	El nombre del parámetro SSM de la imagen.	Cadena	Sí, si `ssmParameterConfigurations` está especificado		
ssmParameterConfigurationsSí, si se especifica ----SEP----:DataType	El tipo de datos del parámetro SSM de la imagen.	Cadena	No		Valores permitidos: text|aws:ec2:image)

Salidas: en la siguiente tabla se incluyen las salidas de esta acción de paso.

Nombre de salida	Description (Descripción)	Tipo
Imágenes configuradas	Una lista de imágenes configuradas.	Matriz

Nombre de salida	Description (Descripción)	Tipo
Imágenes configuradas: ID de cuenta	El ID de cuenta de destino de la imagen distribuida.	Cadena
Imágenes configuradas: nombre	El nombre de la AMI.	Cadena
Imágenes configuradas: un ID	El ID de AMI de la imagen distribuida.	Cadena
Imágenes configuradas: fecha de inicio	Hora UTC en la que comenzó la distribución.	Cadena
Imágenes configuradas: fecha en que se detuvo	Hora UTC en la que se completó la distribución.	Cadena
Imágenes configuradas: paso	Paso en el que se detuvo la distribución.	Completado | | Associate LicensesRunning | UpdateLaunchTemplateRunning | PutSsmParametersRunning | UpdateFastLaunchConfiguration ExportAmiQueued ExportAmiRunning
Imágenes configuradas: región	El tono de la imagen distribuida AWS	Cadena
Imágenes configuradas: estado	Estado de distribución.	Completada | Fallada | Cancelada | TimedOut
Imágenes configuradas: mensaje de error	Mensaje de error, si lo hay.	Cadena

Ejemplo

Especifique la acción de paso en el documento de flujos de trabajo.

```
- name: ApplyImageConfigurations
```

```
  action: ApplyImageConfigurations
  onFailure: Abort
  inputs:
    distributedImages.$: $.stepOutputs.DistributeImageStep.distributedImages
```

Utilice la salida del valor de la acción de paso en el documento de flujos de trabajo.

```
$.stepOutputs.ApplyImageConfigurationsStep.configuredImages
```

BootstrapInstanceForContainer

Esta acción de paso ejecuta un script de servicio para arrancar la instancia con los requisitos mínimos para ejecutar flujos de trabajo de contenedores. Generador de imágenes utiliza la acción sendCommand en la API de Systems Manager para ejecutar este script. Para obtener más información, consulte AWS Systems Manager Run Command.

> **Note**
>
> El script de arranque instala los paquetes AWS CLI y Docker que son requisitos previos para que Image Builder compile correctamente los contenedores de Docker. Si no incluye esta acción de paso, la creación de la imagen podría producir un error.

Tiempo de espera predeterminado: 60 minutos.

Restauración: no hay ninguna restauración para esta acción de paso.

Entradas: en la siguiente tabla se incluyen las entradas admitidas para esta acción de paso.

Nombre de la entrada	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado/a	Restricciones
instanceId	ID de la instancia que se va a arrancar.	Cadena	Sí		Debe ser el ID de la instancia de salida del paso del flujo de trabajo

Nombre de la entrada	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado/a	Restricciones
					que lanzó la instancia para este flujo de trabajo.

Salidas: en la siguiente tabla se incluyen las salidas de esta acción de paso.

Nombre de salida	Description (Descripción)	Tipo
runCommandId	ID de la acción sendCommand de Systems Manager que ejecutó el script de arranque en la instancia.	Cadena
status	Estado devuelto por la acción sendCommand de Systems Manager.	Cadena
salida	Salida devuelta por la acción sendCommand de Systems Manager.	Cadena

Ejemplo

Especifique la acción de paso en el documento de flujos de trabajo.

```
- name: ContainerBootstrapStep
  action: BootstrapInstanceForContainer
  onFailure: Abort
  inputs:
      instanceId.$: $.stepOutputs.LaunchStep.instanceId
```

Utilice la salida del valor de la acción de paso en el documento de flujos de trabajo.

```
$.stepOutputs.ContainerBootstrapStep.status
```

CollectImageMetadata

Esta acción de paso solo es válida para los flujos de trabajo de creación.

EC2 Image Builder ejecuta el agente AWS Systems Manager (Systems Manager) en las EC2 instancias que lanza para crear y probar la imagen. Generador de Imágenes recopila información adicional sobre la instancia utilizada durante la fase de compilación con Systems Manager Inventory. Esta información incluye el nombre y la versión del sistema operativo (OS), así como la lista de paquetes y sus versiones respectivas, según lo indicado por su sistema operativo.

> **Note**
>
> Esta acción de paso solo funciona con las imágenes que se crean AMIs.

Tiempo de espera predeterminado: 30 minutos.

Restauración: Generador de imágenes restaura todos los recursos de Systems Manager que se hayan creado durante este paso.

Entradas: en la siguiente tabla se incluyen las entradas admitidas para esta acción de paso.

Nombre de la entrada	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado/a	Restricciones
instanceId	Instancia de creación en la que se aplicará la configuración de metadatos.	Cadena	Sí		Debe ser el ID de la instancia de salida del paso del flujo de trabajo que lanzó la instancia de creación para este flujo de trabajo.

Salidas: en la siguiente tabla se incluyen las salidas de esta acción de paso.

Nombre de salida	Description (Descripción)	Tipo
osVersion	Nombre y versión del sistema operativo recopilados de la instancia de creación.	Cadena
associationId	ID de asociación de Systems Manager utilizado para la recopilación del inventario.	Cadena

Ejemplo

Especifique la acción de paso en el documento de flujos de trabajo.

```
- name: CollectMetadataStep
  action: CollectImageMetadata
  onFailure: Abort
  inputs:
      instanceId: $.stepOutputs.LaunchStep.instanceId
```

Utilice la salida de la acción de paso en el documento de flujos de trabajo.

```
$.stepOutputs.CollectMetadataStep.osVersion
```

CollectImageScanFindings

Si Amazon Inspector está habilitado para la cuenta y el análisis de imágenes está habilitado para la canalización, este paso de acción recopila los resultados del análisis de imágenes notificados por Amazon Inspector para la instancia de prueba. Esta acción de paso no está disponible para los flujos de trabajo de creación.

Tiempo de espera predeterminado: 120 minutos.

Restauración: no hay ninguna restauración para esta acción de paso.

Entradas: en la siguiente tabla se incluyen las entradas admitidas para esta acción de paso.

Nombre de la entrada	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado/a	Restricciones
instanceId	ID de la instancia en la que se ejecutó el análisis.	Cadena	Sí		Debe ser el ID de la instancia de salida del paso del flujo de trabajo que lanzó la instancia para este flujo de trabajo.

Salidas: en la siguiente tabla se incluyen las salidas de esta acción de paso.

Nombre de salida	Description (Descripción)	Tipo
runCommandId	ID de la acción sendCommand de Systems Manager que ejecutó el script para recopilar los resultados.	Cadena
status	Estado devuelto por la acción sendCommand de Systems Manager.	Cadena
salida	Salida devuelta por la acción sendCommand de Systems Manager.	Cadena

Ejemplo

Especifique la acción de paso en el documento de flujos de trabajo.

```
- name: CollectFindingsStep
  action: CollectImageScanFindings
  onFailure: Abort
  inputs:
      instanceId.$: $.stepOutputs.LaunchStep.instanceId
```

Utilice la salida del valor de la acción de paso en el documento de flujos de trabajo.

```
$.stepOutputs.CollectFindingsStep.status
```

CreateImage

Esta acción de paso crea una imagen a partir de una instancia en ejecución con la EC2 `CreateImage` API de Amazon. Durante el proceso de creación, la acción de paso espera lo necesario para comprobar que los recursos han alcanzado el estado correcto antes de continuar.

Tiempo de espera predeterminado: 720 minutos.

Restauración: no hay ninguna restauración para esta acción de paso.

Entradas: en la siguiente tabla se incluyen las entradas admitidas para esta acción de paso.

Nombre de la entrada	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado/a	Restricciones
instanceId	Instancia desde la que se va a crear la nueva imagen.	Cadena	Sí		La instancia del ID de instancia proporcionado debe encontrarse en estado `running` cuando se inicie este paso.

Salidas: en la siguiente tabla se incluyen las salidas de esta acción de paso.

Nombre de salida	Description (Descripción)	Tipo
imageId	ID de la AMI de la imagen que se crea.	Cadena

Ejemplo

Especifique la acción de paso en el documento de flujos de trabajo.

```
- name: CreateImageFromInstance
  action: CreateImage
  onFailure: Abort
  inputs:
      instanceId.$: "i-1234567890abcdef0"
```

Utilice la salida del valor de la acción de paso en el documento de flujos de trabajo.

```
$.stepOutputs.CreateImageFromInstance.imageId
```

DistributeImage

Esta acción de paso distribuye una AMI a regiones y cuentas específicas. Crea copias de la AMI en las regiones y cuentas de destino en función de la configuración de distribución proporcionada en las solicitudes CreateImage CreateImagePipeline APIs o de la configuración de distribución personalizada proporcionada en el flujo de trabajo para anular la configuración de distribución.

Tiempo de espera predeterminado: 360 minutos

Tiempo de espera máximo: 720 minutos

Restauración: no hay ninguna restauración para esta acción de paso.

Entradas: en la siguiente tabla se incluyen las entradas admitidas para esta acción de paso.

Nombre de la entrada	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado/a	Restricciones
region	La lista de regiones para	Cadena	Sí		Longitud mínima de 1.

Nombre de la entrada	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado/a	Restricciones
	distribuir la imagen.				La longitud máxima es 1024.
name	El nombre de la configuración de distribución.	Cadena	No		
description	Las distribuciones de la configuración de distribución.	Cadena	No		
targetAccountIds	Cuenta IDs a la que se va a distribuir la imagen.	Matriz	No		
Ami Tags	Las etiquetas de la configuración de distribución.	Asignación	No		
kmsKeyId	Claves de KMS que se van a aplicar a la imagen distribuida.	Cadena	No		

Salidas: en la siguiente tabla se incluyen las salidas de esta acción de paso.

Nombre de salida	Description (Descripción)	Tipo
Imágenes distribuidas	Una lista de imágenes distribuidas	Matriz
Imágenes distribuidas: región	La AWS región de la imagen distribuida.	Cadena
Imágenes distribuidas: nombre	El nombre de la AMI.	Cadena
Imágenes distribuidas: un MIID	El ID de AMI de la imagen distribuida.	Cadena
Imágenes distribuidas: ID de cuenta	El ID de cuenta de destino de la imagen distribuida.	Cadena
Imágenes distribuidas: fecha de inicio	Hora UTC en la que comenzó la distribución.	Cadena
Imágenes distribuidas: fecha en la que se detuvo	Hora UTC en la que se completó la distribución.	Cadena
Imágenes distribuidas: estado	Estado de distribución.	Completada | Fallada | Cancelada | TimedOut
Imágenes distribuidas: paso 2	Paso en el que se detuvo la distribución.	Completado| CopyAmiRunning
Imágenes distribuidas: mensaje de error	Mensaje de error, si lo hay.	Cadena

Ejemplo

Especifique la acción de paso en el documento de flujos de trabajo.

```
- name: DistributeImage
  action: DistributeImage
  onFailure: Abort
  inputs:
```

```
    distributions:
      - region.$: "$.parameters.SourceRegion"
        description: "AMI distribution to source region"
        amiTags:
          DistributionTest: "SourceRegion"
          WorkflowStep: "DistributeToSourceRegion"
          BuildDate: "{{imagebuilder:buildDate:yyyyMMHHss}}"
          BuildVersion: "{{imagebuilder:buildVersion}}"
```

Utilice la salida del valor de la acción de paso en el documento de flujos de trabajo.

```
$.stepOutputs.DistributeImageStep.distributedImages
```

ExecuteComponents

Esta acción de paso ejecuta los componentes que se especifican en la receta de la imagen actual que se está creando. Los flujos de trabajo de creación ejecutan los componentes de creación en la instancia de creación. Los flujos de trabajo de prueba solo ejecutan los componentes de prueba en la instancia de prueba.

Generador de imágenes utiliza la acción sendCommand en la API de Systems Manager para ejecutar los componentes. Para obtener más información, consulte AWS Systems Manager Run Command.

Tiempo de espera predeterminado: 720 minutos.

Restauración: no hay ninguna restauración para esta acción de paso.

Entradas: en la siguiente tabla se incluyen las entradas admitidas para esta acción de paso.

Nombre de la entrada	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado/a	Restricciones
instanceId	ID de la instancia en la que deben ejecutarse los componentes.	Cadena	Sí		Debe ser el ID de la instancia de salida del paso del flujo de trabajo

Nombre de la entrada	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado/a	Restricciones
					que lanzó la instancia para este flujo de trabajo.

Salidas: en la siguiente tabla se incluyen las salidas de esta acción de paso.

Nombre de salida	Description (Descripción)	Tipo
runCommandId	ID de la acción sendCommand de Systems Manager que ejecutó los componentes en la instancia.	Cadena
status	Estado devuelto por la acción sendCommand de Systems Manager.	Cadena
salida	Salida devuelta por la acción sendCommand de Systems Manager.	Cadena

Ejemplo

Especifique la acción de paso en el documento de flujos de trabajo.

```
- name: ExecComponentsStep
  action: ExecuteComponents
  onFailure: Abort
  inputs:
      instanceId: $.stepOutputs.LaunchStep.instanceId
```

Utilice la salida de la acción de paso en el documento de flujos de trabajo.

```
$.stepOutputs.ExecComponentsStep.status
```

ExecuteStateMachine

Esta acción de paso inicia la ejecución de una máquina de AWS Step Functions estados desde un flujo de trabajo de Image Builder. Image Builder utiliza la `StartExecution` API Step Functions para iniciar la máquina de estados y espera a que finalice. Esto resulta útil para integrar flujos de trabajo complejos, procesos de validación de conformidad o certificación en su proceso de creación de imágenes.

Para obtener más información, consulte Más información sobre las máquinas de estado en Step Functions de la Guía para AWS Step Functions desarrolladores.

Tiempo de espera predeterminado: 6 horas

Tiempo de espera máximo: 24 horas

Restauración: no hay ninguna restauración para esta acción de paso.

Entradas: en la siguiente tabla se incluyen las entradas admitidas para esta acción de paso.

Nombre de la entrada	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado/a	Restricciones
stateMachineArn	El ARN de la máquina de estado de Step Functions que se va a ejecutar.	Cadena	Sí		Debe ser un ARN de máquina de estado válido.
input	Datos de entrada JSON para proporcionarlos a la máquina de estados.	Cadena	No	{}	Debe ser una cadena JSON válida, longitud máxima: 16 KiB.

Salidas: en la siguiente tabla se incluyen las salidas de esta acción de paso.

Nombre de salida	Description (Descripción)	Tipo
executionArn	El ARN de la ejecución de la máquina de estados.	Cadena

Se requieren permisos de IAM

Su función de ejecución personalizada debe tener los siguientes permisos para utilizar esta acción escalonada:

Permitir acciones

- `states:StartExecution`
- `states:DescribeExecution`

Especifique los recursos

- `arn:aws:states:us-west-2:111122223333:stateMachine:state-machine-name`
- `arn:aws:states:us-west-2:111122223333:execution:state-machine-name:*`

Ejemplo

Especifique la acción de paso en el documento de flujos de trabajo.

```
- name: ValidateImageCompliance
  action: ExecuteStateMachine
  timeoutSeconds: 3600
  onFailure: Abort
  inputs:
    stateMachineArn: arn:aws:states:us-
west-2:111122223333:stateMachine:ImageComplianceValidation
    input: |
      {
        "imageId": "{{ $.stepOutputs.CreateImageFromInstance.imageId }}",
        "region": "us-west-2",
        "complianceLevel": "high",
        "requiredScans": ["cve", "benchmark", "configuration"]
```

```
    }
```

Utilice la salida del valor de la acción de paso en el documento de flujos de trabajo.

```
$.stepOutputs.ValidateImageCompliance.executionArn
```

LaunchInstance

Esta acción de paso lanza una instancia en su Cuenta de AWS cuenta y espera a que el agente de Systems Manager se ejecute en la instancia antes de pasar al siguiente paso. La acción de lanzamiento utiliza las opciones de la receta y los recursos de configuración de la infraestructura asociados a la imagen. Por ejemplo, el tipo de instancia que se va a lanzar proviene de la configuración de la infraestructura. La salida es el ID de la instancia que se lanzó.

La entrada `waitFor` configura la condición que cumple con el requisito de finalización del paso.

Tiempo de espera predeterminado: 60 minutos.

Restauración: en el caso de las instancias de creación, la restauración realiza la acción que se configuró en el recurso de configuración de la infraestructura. De forma predeterminada, las instancias de creación se terminan si se produce un error en la creación de la imagen. Sin embargo, hay una opción en la configuración de la infraestructura para conservar la instancia de creación para la resolución de problemas.

Entradas: en la siguiente tabla se incluyen las entradas admitidas para esta acción de paso.

Nombre de la entrada	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado/a	Restricciones
imageIdOverride	La imagen que se utilizará para lanzar la instancia	Cadena	No	Etapa de construcción: imagen, receta, imagen base Etapa de prueba: AMI de salida desde la	Debe ser un ID de AMI válido

Nombre de la entrada	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado/a	Restricciones
				etapa de compilación	
instanceTypesOverride	Image Builder prueba cada tipo de instancia de la lista hasta encontrar uno que se inicie correctamente.	Lista de cadena	No	Tipos de instancias especificados en la configuración de su infraestructura	Deben ser tipos de instancia válidos
waitFor	La condición que hay que esperar antes de completar el paso del flujo de trabajo y pasar al siguiente paso	Cadena	Sí		Generador de Imágenes es compatible con `ssmAgent`.

Salidas: en la siguiente tabla se incluyen las salidas de esta acción de paso.

Nombre de salida	Description (Descripción)	Tipo
instanceId	ID de la instancia que se lanzó.	Cadena

Ejemplo

Especifique la acción de paso en el documento de flujos de trabajo.

```
- name: LaunchStep
  action: LaunchInstance
  onFailure: Abort
  inputs:
    waitFor: ssmAgent
```

Utilice la salida de la acción de paso en el documento de flujos de trabajo.

```
$.stepOutputs.LaunchStep.instanceId
```

ModifyImageAttributes

Esta acción de paso modifica los atributos de la distribución AMIs, como los permisos de lanzamiento y otros atributos de la AMI. Funciona sobre los AMIs que se han distribuido a las regiones y cuentas de destino.

Tiempo de espera predeterminado: 120 minutos.

Tiempo de espera máximo: 180 minutos

Restauración: no hay ninguna restauración para esta acción de paso.

Entradas: en la siguiente tabla se incluyen las entradas admitidas para esta acción de paso.

Nombre de la entrada	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado/a	Restricciones
region	La región de la imagen.	Cadena	Sí		
Permiso de lanzamiento			No		
Permiso de inicio: ID de usuario	El usuario que debe modificar IDs los permisos de lanzamiento de la imagen.	Cadena	No		

Nombre de la entrada	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado/a	Restricciones
Permiso de lanzamiento: grupos de usuarios	Los grupos de usuarios que se van a modificar en los permisos de lanzamiento de la imagen.	Cadena	No		
Permiso de lanzamiento: ARN de la organización	La AWS organización que se va ARNs a modificar en los permisos de lanzamiento de la imagen.	Cadena	No		
Permiso de lanzamiento: organizationalUnitArns	La unidad AWS organizativa ARNs que se modificará en los permisos de lanzamiento de la imagen.	Cadena	No		

Salidas: en la siguiente tabla se incluyen las salidas de esta acción de paso.

Nombre de salida	Description (Descripción)	Tipo
Imágenes modificadas	Una lista de imágenes modificadas	Matriz
Imágenes modificadas: ID de cuenta	El ID de cuenta de destino de la imagen distribuida.	Cadena
Imágenes modificadas: nombre	El nombre de la AMI.	Cadena
Imágenes modificadas: un ID	El ID de AMI de la imagen distribuida.	Cadena
Imágenes modificadas: fecha de inicio	Hora UTC en la que comenzó la distribución.	Cadena
Imágenes modificadas: fecha de interrupción	Hora UTC en la que se completó la distribución.	Cadena
Imágenes modificadas: paso	Paso en el que se detuvo la distribución.	Completado| ModifyAmi Running
Imágenes modificadas: región	La AWS región de la imagen.	Cadena
Imágenes modificadas: estado	Estado de distribución.	Completada | Fallada | Cancelada | TimedOut
Imágenes modificadas: mensaje de error	Mensaje de error, si lo hay.	Cadena

Ejemplo

Especifique la acción de paso en el documento de flujos de trabajo.

```
- name: ModifyImageAttributes
  action: ModifyImageAttributes
  onFailure: Abort
  inputs:
```

```
    distributedImages.$: $.stepOutputs.DistributeImageStep.distributedImages
```

Utilice la salida del valor de la acción de paso en el documento de flujos de trabajo.

```
$.stepOutputs.ModifyImageAttributesStep.modifiedImages
```

RegisterImage

Esta acción de paso registra una nueva imagen de máquina de Amazon (AMI) mediante la EC2 RegisterImage API de Amazon. Le permite crear una AMI a partir de una instantánea o conjunto de instantáneas existentes, especificando varios atributos de imagen.

Tiempo de espera predeterminado: 720 minutos.

Restauración: no hay ninguna restauración para esta acción de paso.

Entradas: en la siguiente tabla se incluyen las entradas admitidas para esta acción de paso.

Nombre de la entrada	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado/a	Restricciones
architecture	La arquitectura de la AMI.	Cadena	No		Valores válidos: i386, x86_64, arm64, x86_64_mac, arm64_mac
blockDeviceMapping	Las entradas de mapeo de dispositivos de bloques para la AMI.	Matriz	No		
Modo de arranque	El modo de arranque de la AMI.	Cadena	No		Valores válidos: legacy-bios, uefi, uefi-preferred
description	Descripción de la AMI.	Cadena	No		

Nombre de la entrada	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado/a	Restricciones
enaSupport	Si están habilitadas las redes mejoradas con ENA.	Booleano	No		
Ubicación de la imagen	La ubicación del manifiesto de la AMI.	Cadena	No		Necesario para versiones respaldadas por S3 AMIs
Soporte para iMDS	El nivel de IMDSv2 soporte.	Cadena	No		Valores válidos: v2.0

Nombre de la entrada	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado/a	Restricciones
includeSnapshotTags	Si se deben incluir las etiquetas de la primera instantánea definida en el mapeo de dispositivos de bloques.	Booleano	No	FALSO	Si se establece en true, las etiquetas se incluyen de la siguiente manera: • Las etiquetas del primer volumen `SnapshotId` de EBS de la `blockDeviceMapping` lista que contiene un `SnapshotId` se combinan con las etiquetas de registro de la AMI. • Las etiquetas de registro AMI tienen prioridad sobre las etiquetas de instantáneas con la misma clave. • AWS las etiquetas reservadas (aquellas cuyas claves comienzan por`aws:`) se excluyen automáticamente. •

Nombre de la entrada	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado/a	Restricciones
					Si `SnapshotId` se definen varios volúmenes de EBS con, solo se incluyen las etiquetas del primer volumen de EBS de la lista que contenga `SnapshotId` un.
ID del núcleo	El ID del núcleo que se va a utilizar.	Cadena	No		
ID del disco RAM	El ID del disco RAM que se va a utilizar.	Cadena	No		
rootDeviceName	El nombre del dispositivo raíz.	Cadena	No		Ejemplo: /dev/sda1
sriovNetSupport	Redes mejoradas con la interfaz Intel 82599 VF.	Cadena	No		
Soporte TPM	Compatibilidad con la versión TPM.	Cadena	No		Valores válidos: v2.0
Datos UEFI	Datos UEFI codificados en Base64.	Cadena	No		

Nombre de la entrada	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado/a	Restricciones
Tipo de virtualización	El tipo de virtualización.	Cadena	No		Valores válidos: hvm, paravirtual

Salidas: en la siguiente tabla se incluyen las salidas de esta acción de paso.

Nombre de salida	Description (Descripción)	Tipo
imageId	El ID de AMI de la imagen registrada.	Cadena

Ejemplo

Especifique la acción de paso en el documento de flujos de trabajo.

```
- name: RegisterNewImage
  action: RegisterImage
  onFailure: Abort
  inputs:
    architecture: "x86_64"
    bootMode: "uefi"
    blockDeviceMapping:
      - DeviceName: "/dev/sda1"
        Ebs:
          SnapshotId: "snap-1234567890abcdef0"
          VolumeSize: 100
          VolumeType: "gp3"
    rootDeviceName: "/dev/sda1"
    virtualizationType: "hvm"
```

Utilice la salida del valor de la acción de paso en el documento de flujos de trabajo.

```
$.stepOutputs.RegisterNewImage.imageId
```

Ejemplo con etiquetas SnapshotId from another step y snapshot incluidas en la AMI generada

```
- name: CreateSnapshot
  action: RunCommand
  onFailure: Abort
  inputs:
    instanceId: "i-1234567890abcdef0"
    documentName: "AWS-RunShellScript"
    parameters:
      commands:
        - "aws ec2 create-snapshot --volume-id vol-1234567890abcdef0 --description
 'Snapshot for AMI' --query 'SnapshotId' --output text"

- name: RegisterImageFromSnapshot
  action: RegisterImage
  onFailure: Abort
  inputs:
    architecture: "x86_64"
    bootMode: "uefi"
    blockDeviceMapping:
      - DeviceName: "/dev/sda1"
        Ebs:
          SnapshotId.$: "$.stepOutputs.CreateSnapshot.output[0]"
          VolumeSize: 100
          VolumeType: "gp3"
    includeSnapshotTags: true
    rootDeviceName: "/dev/sda1"
    virtualizationType: "hvm"
```

RunCommand

Esta acción de paso ejecuta un documento de comandos para el flujo de trabajo. Generador de imágenes utiliza la acción sendCommand en la API de Systems Manager para ejecutarlo automáticamente. Para obtener más información, consulte AWS Systems Manager Run Command.

Tiempo de espera predeterminado: 12 horas.

Restauración: no hay ninguna restauración para esta acción de paso.

Entradas: en la siguiente tabla se incluyen las entradas admitidas para esta acción de paso.

Nombre de la entrada	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado/a	Restricciones
instanceId	ID de la instancia en la que se va a ejecutar el documento de comandos.	Cadena	Sí		Debe ser el ID de la instancia de salida del paso del flujo de trabajo que lanzó la instancia para este flujo de trabajo.
documentName	Nombre del documento de comandos de Systems Manager que se va a ejecutar.	Cadena	Sí		
parameters	Lista de pares clave-valor para cualquier parámetro que requiera el documento de comandos.	dictionary<string, list<string>>	Condicional		
documentVersion	Versión del documento de comandos	Cadena	No	$DEFAULT	

Nombre de la entrada	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado/a	Restricciones
	que se va a ejecutar.				

Salidas: en la siguiente tabla se incluyen las salidas de esta acción de paso.

Nombre de salida	Description (Descripción)	Tipo
runCommandId	ID de la acción sendCommand de Systems Manager que ejecutó el documento de comandos en la instancia.	Cadena
status	Estado devuelto por la acción sendCommand de Systems Manager.	Cadena
salida	Salida devuelta por la acción sendCommand de Systems Manager.	Lista de cadenas

Ejemplo

Especifique la acción de paso en el documento de flujos de trabajo.

```
- name: RunCommandDoc
  action: RunCommand
  onFailure: Abort
  inputs:
    documentName: SampleDocument
    parameters:
        osPlatform:
          - "linux"
    instanceId.$: $.stepOutputs.LaunchStep.instanceId
```

Utilice la salida del valor de la acción de paso en el documento de flujos de trabajo.

```
$.stepOutputs.RunCommandDoc.status
```

RunSysPrep

Esta acción de paso utiliza la acción sendCommand en la API de Systems Manager para ejecutar el documento `AWSEC2-RunSysprep` para las instancias de Windows antes de que la instancia de creación se cierre para la instantánea. Estas acciones siguen las prácticas AWS recomendadas para endurecer y limpiar la imagen.

Tiempo de espera predeterminado: 60 minutos.

Restauración: no hay ninguna restauración para esta acción de paso.

Entradas: en la siguiente tabla se incluyen las entradas admitidas para esta acción de paso.

Nombre de la entrada	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado/a	Restricciones
instanceId	ID de la instancia en la que se va a ejecutar el documento `AWSEC2-RunSysprep` .	Cadena	Sí		Debe ser el ID de la instancia de salida del paso del flujo de trabajo que lanzó la instancia para este flujo de trabajo.

Salidas: en la siguiente tabla se incluyen las salidas de esta acción de paso.

Nombre de salida	Description (Descripción)	Tipo
runCommandId	ID de la acción sendCommand de Systems Manager que ejecutó el documento	Cadena

Nombre de salida	Description (Descripción)	Tipo
	`AWSEC2-RunSysprep` en la instancia.	
status	Estado devuelto por la acción sendCommand de Systems Manager.	Cadena
salida	Salida devuelta por la acción sendCommand de Systems Manager.	Cadena

Ejemplo

Especifique la acción de paso en el documento de flujos de trabajo.

```
- name: RunSysprep
  action: RunSysPrep
  onFailure: Abort
  inputs:
      instanceId.$: $.stepOutputs.LaunchStep.instanceId
```

Utilice la salida del valor de la acción de paso en el documento de flujos de trabajo.

```
$.stepOutputs.RunSysprep.status
```

SanitizeInstance

Esta acción de paso ejecuta el script de saneamiento recomendado para las instancias de Linux antes de que la instancia de creación se cierre para la instantánea. El script de saneamiento ayuda a garantizar que la imagen final siga las prácticas recomendadas de seguridad y que se eliminen los artefactos u opciones de creación que no deban transferirse a la instantánea. Para obtener más información sobre el script, consulte Limpieza posterior a la creación necesaria. Esta acción de paso no se aplica a las imágenes de contenedor.

Generador de imágenes utiliza la acción sendCommand en la API de Systems Manager para ejecutar este script. Para obtener más información, consulte AWS Systems Manager Run Command.

Tiempo de espera predeterminado: 60 minutos.

Restauración: no hay ninguna restauración para esta acción de paso.

Entradas: en la siguiente tabla se incluyen las entradas admitidas para esta acción de paso.

Nombre de la entrada	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado/a	Restricciones
instanceId	ID de la instancia que se va a sanear.	Cadena	Sí		Debe ser el ID de la instancia de salida del paso del flujo de trabajo que lanzó la instancia para este flujo de trabajo.

Salidas: en la siguiente tabla se incluyen las salidas de esta acción de paso.

Nombre de salida	Description (Descripción)	Tipo
runCommandId	ID de la acción sendCommand de Systems Manager que ejecutó el script de saneamiento en la instancia.	Cadena
status	Estado devuelto por la acción sendCommand de Systems Manager.	Cadena
salida	Salida devuelta por la acción sendCommand de Systems Manager.	Cadena

Ejemplo

Especifique la acción de paso en el documento de flujos de trabajo.

```
- name: SanitizeStep
  action: SanitizeInstance
  onFailure: Abort
  inputs:
      instanceId: $.stepOutputs.LaunchStep.instanceId
```

Utilice la salida del valor de la acción de paso en el documento de flujos de trabajo.

```
$.stepOutputs.SanitizeStep.status
```

TerminateInstance

Esta acción de paso termina la instancia con el ID de instancia que se pasa como entrada.

Tiempo de espera predeterminado: 30 minutos.

Restauración: no hay ninguna restauración para esta acción de paso.

Entradas: en la siguiente tabla se incluyen las entradas admitidas para esta acción de paso.

Nombre de la entrada	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado/a	Restricciones
instanceId	ID de la instancia que se va a terminar.	Cadena	Sí		

Salidas: no hay salidas para esta acción de paso.

Ejemplo

Especifique la acción de paso en el documento de flujos de trabajo.

```
- name: TerminateInstance
  action: TerminateInstance
  onFailure: Continue
  inputs:
```

```
      instanceId.$: i-1234567890abcdef0
```

WaitForAction

Esta acción de paso pausa el flujo de trabajo en ejecución y espera a recibir una acción externa de la acción SendWorkflowStepAction de la API de Generador de imágenes. Este paso publica un EventBridge evento en el bus de EventBridge eventos predeterminado con un tipo `EC2 Image Builder Workflow Step Waiting` de detalle. El paso también puede enviar una notificación de SNS si proporciona un ARN de tema de SNS o invocar una función de Lambda de forma asíncrona si proporciona un nombre de función de Lambda.

Tiempo de espera predeterminado: 3 días.

Restauración: no hay ninguna restauración para esta acción de paso.

Entradas: en la siguiente tabla se incluyen las entradas admitidas para esta acción de paso.

Nombre de la entrada	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado/a	Restricciones
snsTopicArn	ARN de un tema de SNS opcional al que enviar una notificación cuando el paso del flujo de trabajo esté pendiente.	Cadena	No		
lambdaFunctionName	Un nombre o ARN opcional de la función Lambda para invocar de forma asíncrona cuando el	Cadena	No		

Nombre de la entrada	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado/a	Restricciones
	paso del flujo de trabajo esté pendiente.				
payload	Cadena JSON utilizada como mensaje para SNS y carga útil para Lambda. Si se proporciona, se incluye una carga útil personalizada de forma predeterminada. message/payload, used for SNS and Lambda respectively. If not provided, generates default message/payload	Cadena	No		Debe ser una cadena JSON válida, máximo 16 KiB

Salidas: en la siguiente tabla se incluyen las salidas de esta acción de paso.

Nombre de salida	Description (Descripción)	Tipo
acción	Acción que devuelve la acción SendWorkflowStepAction de la API.	Cadena (`RESUME` o `STOP`)
razón	Motivo de la acción devuelta.	Cadena

Ejemplo

Especifique la acción del paso en el documento de flujo de trabajo con la notificación de SNS.

```
- name: SendEventAndWait
  action: WaitForAction
  onFailure: Abort
  inputs:
    snsTopicArn: arn:aws:sns:us-west-2:111122223333:ExampleTopic
```

Especifique la acción del paso en el documento de flujo de trabajo con la invocación de la función Lambda.

```
- name: SendEventAndWaitWithLambda
  action: WaitForAction
  onFailure: Abort
  inputs:
    lambdaFunctionName: ExampleFunction
    payload: |
      {
        "imageId": "{{ $.stepOutputs.CreateImageFromInstance.imageId }}",
        "region": "us-west-2"
      }
```

Utilice la salida del valor de la acción de paso en el documento de flujos de trabajo.

```
$.stepOutputs.SendEventAndWait.reason
```

WaitForSSMAgent

Esta acción escalonada espera a que una EC2 instancia se pueda administrar AWS Systems Manager después de los períodos esperados de falta de respuesta. Resulta especialmente útil para los flujos de trabajo con interrupciones conocidas de la instancia, como los reinicios del sistema, las actualizaciones del sistema operativo o las operaciones específicas de la plataforma que desconectan temporalmente la instancia del SSM. Image Builder supervisa la instancia hasta que recupera la conectividad SSM o se agota el tiempo de espera.

Tiempo de espera predeterminado: 60 minutos.

Tiempo de espera máximo: 180 minutos

Restauración: no hay ninguna restauración para esta acción de paso.

Entradas: en la siguiente tabla se incluyen las entradas admitidas para esta acción de paso.

Nombre de la entrada	Description (Descripción)	Tipo	Obligatorio/a	Predeterminado/a	Restricciones
instanceId	El ID de la instancia que se va a supervisar para comprobar la conectividad SSM.	Cadena	Sí		Debe ser un ID de EC2 instancia válido

Salidas: en la siguiente tabla se incluyen las salidas de esta acción de paso.

Nombre de salida	Description (Descripción)	Tipo
Status	Estado de conexión del agente SSM.	Cadena

Ejemplo

Especifique la acción de paso en el documento de flujos de trabajo.

```
- name: WaitForInstanceAfterReboot
  action: WaitForSSMAgent
  onFailure: Abort
  timeoutInSeconds: 900 # 15 minutes
  inputs:
    instanceId.$: $.stepOutputs.LaunchStep.instanceId
```

Utilice la salida del valor de la acción de paso en el documento de flujos de trabajo.

```
$.stepOutputs.WaitForInstanceAfterReboot.Status
```

Uso de variables dinámicas en el documento de flujos de trabajo

Puede utilizar variables dinámicas en los documentos de flujos de trabajo para representar valores que varían en tiempo de ejecución para el proceso de creación de imágenes. La interpolación de cadenas para variables dinámicas te permite incrustar JSONPath expresiones en contenido estructurado, como cadenas JSON. Esto resulta especialmente útil cuando necesitas pasar valores de tiempo de ejecución dentro de cargas complejas para realizar acciones escalonadas como o. `ExecuteStateMachine` `WaitForAction`

Para usar la interpolación de cadenas para variables dinámicas, coloca las JSONPath expresiones entre corchetes dobles dentro del contenido de la cadena. `"{{...}}"` Solo JSONPath las expresiones envueltas entre corchetes dobles se procesan como variables. Cualquier JSONPath expresión que no esté envuelta entre corchetes dobles se trata literalmente como contenido de cadena.

JSONPath sintaxis de variables de flujo de trabajo dinámico

```
$.<document structure>.[<step name>.]<variable name>
```

Los valores de las variables dinámicas se representan como JSONPath selectores con nodos estructurales que identifican de forma exclusiva la variable objetivo. El primer nodo después de la raíz ($) hace referencia a la estructura del documento de flujos de trabajo; por ejemplo, `stepOutputs` o, en el caso de las variables del sistema de Generador de imágenes, `imageBuilder`. La siguiente lista contiene los nodos de estructura JSONPath de documentos de flujo de trabajo compatibles.

Nodos de estructura de documentos

- parameters: parámetros del flujo de trabajo
- stepOutputs: salidas de un paso del mismo documento de flujos de trabajo
- workflowOutputs: salidas de un documento de flujos de trabajo que ya se ha ejecutado
- imagebuilder: variables del sistema de Generador de imágenes

Los nodos `parameters` y `stepOutputs` de estructura de documentos incluyen un nodo opcional para el nombre del paso. Esto ayuda a garantizar que los nombres de las variables sean únicos en todos los pasos.

El último nodo de JSONPath es el nombre de la variable de destino, por ejemplo`instanceId`.

Cada paso puede hacer referencia al resultado de cualquier acción del paso anterior con estas variables JSONPath dinámicas. Esto también se conoce como encadenamiento o referencia. Para hacer referencia a la salida de una acción de paso anterior, puede utilizar la siguiente variable dinámica.

```
$.stepOutputs.step-name.output-name
```

> **Important**
>
> Cuando un parámetro de entrada hace referencia a una variable dinámica, el indicador de encadenamiento (`.$`) debe estar adjunto al final del nombre del parámetro.

Ejemplo 1: indicador de encadenamiento de parámetros de entrada

El siguiente ejemplo muestra un parámetro de entrada que utiliza la interpolación de cadenas para resolver una variable dinámica en el valor del parámetro en tiempo de ejecución.

```
- name: ApplyTestComponents
  action: ExecuteComponents
  onFailure: Abort
  inputs:
    instanceId.$: "$.stepOutputs.LaunchTestInstance.instanceId"
```

Ejemplo 2: Interpolación de cadenas en variables dinámicas

El siguiente ejemplo demuestra cómo las variables dinámicas utilizan la interpolación de cadenas para determinar los valores en tiempo de ejecución.

```
- name: ValidateImageConfiguration
  action: ExecuteStateMachine
  inputs:
    stateMachineArn: arn:aws:states:us-east-1:111122223333:stateMachine:ImageValidation
    input: |
      {
        "imageId": "{{ $.stepOutputs.CreateImageFromInstance.imageId }}",
        "region": "us-east-1",
        "buildDate": "{{ $.imagebuilder.dateTime }}",
        "instanceType": "{{ $.stepOutputs.LaunchStep.instanceType }}"
      }
```

En este ejemplo, las JSONPath expresiones envueltas entre corchetes dobles se resuelven en tiempo de ejecución:

- `{{ $.stepOutputs.CreateImageFromInstance.imageId }}`- Se resuelve con el ID de imagen real del paso CreateImageFromInstance
- `{{ $.imagebuilder.dateTime }}`- Se resuelve con la marca de tiempo de compilación actual. Consulte Uso de variables del sistema de Generador de imágenes para obtener una lista de las variables del sistema Image Builder que puede utilizar.
- `{{ $.stepOutputs.LaunchStep.instanceType }}`- Se resuelve con el tipo de instancia utilizado en el LaunchStep

Las cadenas literales como, por ejemplo, `"region": "us-east-1"` permanecen sin cambios.

> **Note**
>
> La interpolación de cadenas funciona con cualquier contenido de cadena del documento de flujo de trabajo, incluidas las cadenas multilínea que utilizan el operador pipe () de YAML. `|` El requisito de la corcheta actúa como un mecanismo de escape para distinguir claramente entre JSONPath las variables y el contenido literal del texto.

Uso de variables del sistema de Generador de imágenes

Generador de imágenes proporciona las siguientes variables de sistema que puede utilizar en el documento de flujos de trabajo:

Nombre de variable	Description (Descripción)	Tipo	Ejemplo de valor
cloudWatchLogGrupo	El nombre del grupo de CloudWatch registros para los registros de salida. Formato: `/aws/imagebuilder/<recipe-name>`	Cadena	`/aws/imagebuilder/sampleImageRecipe`
cloudWatchLogTransmisión	El nombre del flujo de CloudWatch registros para los registros de salida.	Cadena	*1.0.0/1*
collectImageMetadata	Configuración que indica a Generador de imágenes si debe recopilar metadatos de la instancia.	Booleano	`true` | `false`
collectImageScanHallazgos	Valor actual de la configuración que permite a Generador de imágenes recopilar los resultados del análisis de imágenes.	Booleano	`true` | `false`

Nombre de variable	Description (Descripción)	Tipo	Ejemplo de valor
imageBuildNumber	Número de versión de creación de la imagen.	Entero	*1*
imageId	ID de AMI de la imagen base.	Cadena	*ami-12345 67890abcdef1*
imageName	El nombre de la imagen.	Cadena	*sampleImage*
imageType	Tipo de salida de la imagen.	Cadena	`AMI` | `Docker`
imageVersionNumber	Número de versión de la imagen.	Cadena	*1.0.0*
instanceProfileName	Nombre del rol de perfil de instancia que Generador de imágenes utiliza para lanzar instancias de creación y prueba.	Cadena	*SampleImageBuilder InstanceProfileRole*
platform	Plataforma del sistema operativo de la imagen que se crea.	Cadena	`Linux` | `Windows` | `MacOS`

Nombre de variable	Description (Descripción)	Tipo	Ejemplo de valor
s3Logs	Objeto JSON que contiene la configuración de los registros de S3 que escribe Generador de imágenes.	Objeto JSON	{'S3logs': {'s3 BucketName ':', 's3 *sample-bucket* ':KeyPrefix' '}} *ib-logs*
securityGroups	El grupo de seguridad IDs que se aplica a las instancias de creación y prueba.	Lista [cadena]	*[sg-12345 67890abcd ef1, sg-111122 22333344445]*
sourceImageARN	Nombre de recurso de Amazon (ARN) del recurso de imagen de Generador de imágenes que el flujo de trabajo utiliza para las etapas de creación y prueba.	Cadena	arn:aws:imagebuild er: ::image//*us-east-1 111122223 333 sampleIma ge 1.0.0/1*
subnetId	ID de la subred en la que se lanzan las instancias de creación y prueba.	Cadena	*subnet-12 34567890a bcdef1*

Nombre de variable	Description (Descripción)	Tipo	Ejemplo de valor
terminateInstanceOnFallo	Valor actual de la configuración que indica a Generador de imágenes que termine la instancia en caso de error o que la conserve para la solución de problemas.	Booleano	`true` | `false`
workflowPhase	Etapa actual que se ejecuta para la ejecución del flujo de trabajo.	Cadena	`Build` | `Test`
workingDirectory	Ruta al directorio de trabajo.	Cadena	`/tmp`

Uso de instrucciones condicionales en los pasos del flujo de trabajo

Las instrucciones condicionales comienzan con el atributo `if` del documento de instrucciones. El objetivo final de la instrucción `if` es determinar si se debe ejecutar la acción del paso u omitirla. Si la instrucción `if` se resuelve en `true`, se ejecuta la acción del paso. Si se resuelve en `false`, Generador de imágenes omite la acción del paso y registra el estado del paso `SKIPPED` en el registro.

La instrucción `if` admite las instrucciones de ramificación (`and`, `or`) y modificadores condicionales (`not`). También admite los siguientes operadores de comparación que realizan comparaciones de valores (igual, menor que, mayor que) en función de los tipos de datos que compara (cadena o número).

Operadores de comparación compatibles

- `booleanEquals`
- `numberEquals`
- `numberGreaterThan`
- `numberGreaterThanEquals`
- `numberLessThan`
- `numberLessThanEquals`
- `stringEquals`

Reglas para las instrucciones de ramificación y modificadores condicionales

Se aplican las reglas siguientes para las instrucciones de ramificación (`and`, `or`) y modificadores condicionales (`not`).

- Las instrucciones de ramificación y los modificadores condicionales deben aparecer en una línea por sí mismos.
- Las instrucciones de ramificación y los modificadores condicionales deben seguir reglas de nivel.
  - Solo puede haber una instrucción en el nivel principal.
  - Cada rama o modificador secundario inicia un nuevo nivel.

  Para obtener más información sobre los niveles, consulte Niveles anidados en instrucciones condicionales.
- Cada instrucción de ramificación debe tener al menos una instrucción condicional secundaria, pero no más de diez.
- Los modificadores condicionales funcionan solo en una instrucción condicional secundaria.

Niveles anidados en instrucciones condicionales

Las instrucciones condicionales funcionan en varios niveles en una sección propia. Por ejemplo, el atributo `if` de la instrucción aparece en el mismo nivel del documento de flujos de trabajo que el nombre y la acción del paso. Esta es la base de la instrucción condicional.

Puede especificar hasta cuatro niveles de instrucciones condicionales, pero solo puede aparecer una instrucción en el nivel principal. A todas las demás instrucciones de ramificación, modificadores

condicionales u operadores condicionales se les aplica una sangría a partir de ahí (una sangría por nivel).

En el siguiente esquema se muestra el número máximo de niveles anidados para una instrucción condicional.

```
base:
  parent:
    - child (level 2)
        - child (level 3)
            child (level 4)
```

Atributo `if`

El atributo `if` especifica la instrucción condicional como atributo del documento. Este es el nivel cero.

Nivel principal

Este es el primer nivel de anidación de las instrucciones condicionales. Solo puede haber una instrucción en este nivel. Si no necesita ramificaciones ni modificadores, puede ser un operador condicional sin instrucciones secundarias. En este nivel no se utiliza la notación de guiones, excepto para los operadores condicionales.

Niveles secundarios

Los niveles del dos al cuatro se consideran niveles secundarios. Las instrucciones secundarias pueden incluir instrucciones ramificadas, modificadores condicionales u operadores condicionales.

Ejemplo: niveles anidados

En el siguiente ejemplo se muestra el número máximo de niveles en una instrucción condicional.

```
if:
  and:                          #first level
    - stringEquals: 'my_string'    #second level
      value: 'my_string'
    - and:                      #also second level
        - numberEquals: '1'        #third level
          value: 1
        - not:                  #also third level
```

```
            stringEquals: 'second_string'      #fourth level
            value: "diff_string"
```

Reglas de anidación

- Cada rama o modificador en el nivel secundario inicia un nuevo nivel.
- A cada nivel se le aplica una sangría.
- Puede haber un máximo de cuatro niveles, incluida una instrucción, modificador u operador en el nivel principal, y hasta tres niveles adicionales.

Ejemplos de instrucción condicional

En este grupo de ejemplos se muestran varios aspectos de las instrucciones condicionales.

Ramificación: and

La instrucción de ramificación `and` se basa en una lista de expresiones que son secundarias a la rama, todas las cuales deben evaluarse como `true`. Generador de imágenes evalúa las expresiones en el orden en que aparecen en la lista. Si alguna expresión se evalúa como `false`, el procesamiento se detiene y la rama se evalúa como `false`.

En el siguiente ejemplo se evalúa como `true`, porque ambas expresiones se evalúan como `true`.

```
if:
  and:
    - stringEquals: 'test_string'
      value: 'test_string'
    - numberEquals: 1
      value: 1
```

Ramificación: or

La instrucción de ramificación `or` se basa en una lista de expresiones que son secundarias de la rama, al menos una de las cuales debe evaluarse como `true`. Generador de imágenes evalúa las expresiones en el orden en que aparecen en la lista. Si alguna expresión se evalúa como `true`, el procesamiento se detiene y la rama se evalúa como `true`.

En el siguiente ejemplo se evalúa como `true`, aunque la primera expresión sea `false`.

```
if:
```

```
  or:
    - stringEquals: 'test_string'
      value: 'test_string_not_equal'
    - numberEquals: 1
      value: 1
```

Modificador condicional: not

El modificador condicional `not` niega las instrucciones condicionales que son secundarias de la rama.

En el siguiente ejemplo, se evalúa como `true` cuando el modificador `not` niega la instrucción condicional `stringEquals`.

```
if:
  not:
    - stringEquals: 'test_string'
      value: 'test_string_not_equal'
```

Instrucción condicional: booleanEquals

El operador de comparación `booleanEquals` compara valores booleanos y devuelve el valor true si los valores booleanos coinciden exactamente.

En el siguiente ejemplo se determina si se ha habilitado `collectImageScanFindings`.

```
if:
  - booleanEquals: true
    value: '$.imagebuilder.collectImageScanFindings'
```

Instrucción condicional: stringEquals

El operador de comparación `stringEquals` compara dos cadenas y devuelve el valor true si las cadenas coinciden exactamente. Si alguno de los valores no es una cadena, Generador de imágenes lo convierte en una cadena antes de compararlos.

En el siguiente ejemplo, se compara la variable de sistema de la plataforma para determinar si el flujo de trabajo se ejecuta en una plataforma Linux.

```
if:
  - stringEquals: 'Linux'
```

```
    value: '$.imagebuilder.Platform'
```

Instrucción condicional: numberEquals

El operador de comparación `numberEquals` compara dos números y devuelve el valor true si los números son iguales. Los números que se van a comparar deben tener uno de los siguientes formatos.

- Entero
- Flotante
- Cadena que coincide con el siguiente patrón de expresiones regulares: `^-?[0-9]+(\.)?[0-9]+$`.

Todas las comparaciones del siguiente ejemplo se evalúan como `true`.

```
if:
  # Value provider as a number
  numberEquals: 1
  value: '1'

  # Comparison value provided as a string
  numberEquals: '1'
  value: 1

  # Value provided as a string
  numberEquals: 1
  value: '1'

  # Floats are supported
  numberEquals: 5.0
  value: 5.0

  # Negative values are supported
  numberEquals: -1
  value: -1
```

Administración de la creación de imágenes personalizadas en Generador de imágenes mediante un proceso de canalización repetible

Las canalizaciones de imágenes de Image Builder proporcionan un marco de automatización para crear y mantener imágenes personalizadas AMIs y de contenedores. Las canalizaciones ofrecen las siguientes funciones:

- Reúna la imagen base, cree y pruebe los componentes para la personalización de la imagen, la configuración de la infraestructura y los ajustes de distribución.
- Facilitar la programación de los procesos de mantenimiento automatizados mediante `Schedule builder` en el asistente de la consola o ingresando expresiones cron para las actualizaciones periódicas de las imágenes.
- Habilitar la detección de cambios en la imagen base y los componentes para omitir automáticamente las compilaciones programadas cuando no haya cambios.
- Habilite la automatización basada en reglas a través de Amazon. EventBridge

> **Note**
>
> Para obtener más información sobre el uso de la EventBridge API para ver o cambiar las reglas, consulta la referencia de la EventBridge API de Amazon. Para obtener más información sobre el uso de EventBridge events comandos en la AWS CLI para ver o cambiar las reglas, consulte los eventos en la Referencia de AWS CLI comandos.

Temas

- Configurar los ajustes de ejecución de canalizaciones para canalizaciones de imágenes
- Enumeración y visualización de los detalles de la canalización
- Creación y actualización de canalizaciones de imágenes de AMI
- Creación y actualización de canalizaciones de imágenes de contenedor
- Configuración de flujos de trabajo de canalizaciones de imágenes en Generador de imágenes
- Usa EventBridge reglas con los pipelines de Image Builder

Configurar los ajustes de ejecución de canalizaciones para canalizaciones de imágenes

Puede elegir entre las siguientes opciones para programar la ejecución de la canalización:

Generador de horarios

Utilice el Schedule Builder para configurar la ejecución automática y recurrente de las canalizaciones. Tú defines cuándo y con qué frecuencia debe ejecutarse tu canalización (día, hora y frecuencia). La programación predeterminada es todas las semanas en función del día y la hora en que se creó la programación (UTC).

Expresión rate

Ejecuta la canalización automáticamente con una expresión cron que especifique la programación. Para obtener más información sobre la sintaxis cron que utiliza Image Builder, consulteUtilizar expresiones cron en Generador de imágenes.

Manual

La canalización no se ejecuta según una programación. En la consola, selecciona Ejecutar canalización en el menú Acciones para ejecutar la canalización. Desde el AWS CLI, puedes correr`start-image-pipeline-execution`.

Configuración de dependencias

En el caso de las compilaciones programadas, puedes elegir si deseas ejecutarlas siempre según lo programado o saltarte la ejecución en proceso, a menos que haya actualizaciones de dependencias, como un cambio en la imagen base o en un componente que se utilice en la receta.

Inhabilita automáticamente una canalización defectuosa

Para las canalizaciones de imágenes que se ejecutan según una programación, puede configurar el número máximo de errores consecutivos que se deben permitir (hasta`10`) antes de que Image Builder desactive automáticamente la canalización.

Desactivar automáticamente los ajustes

Image Builder realiza un seguimiento del número de errores consecutivos de las ejecuciones de canalizaciones programadas y realiza una de las siguientes acciones cada vez que se ejecuta de forma programada:

- Si la ejecución de la canalización se realiza correctamente, el número de errores consecutivos se restablece a cero.
- Si se produce un error en la ejecución de la canalización, Image Builder incrementa el número de errores consecutivos. Si el recuento de errores supera el límite definido en`AutoDisablePolicy`, Image Builder deshabilita la canalización.

El recuento de errores consecutivos también se restablece a cero en las siguientes condiciones:

- La canalización se ejecuta manualmente y se ejecuta correctamente.
- Se actualiza la configuración de la canalización.

Si la canalización se ejecuta manualmente y falla, el recuento sigue siendo el mismo. La siguiente ejecución programada continúa incrementándose donde la dejó antes.

Configure el registro de canalizaciones

Al crear o actualizar una canalización de imágenes, puede configurar grupos de CloudWatch registros personalizados para los registros de creación y canalización de imágenes. Asegúrese de que su función de ejecución de canalización personalizada tenga los siguientes permisos para crear y acceder a los recursos del grupo de registros.

- registros: CreateLogGroup
- registros: CreateLogStream
- registros: PutLogEvents

Grupos de registros personalizados

Para usar grupos de registros personalizados para la creación de imágenes o la ejecución de canalizaciones, primero cree el grupo de CloudWatch registros en Logs. Para obtener más información, consulte Crear un grupo de CloudWatch registros en la Guía del usuario de Amazon Logs. Para obtener más información sobre los requisitos de denominación de los grupos de CloudWatch registros, consulta CreateLogGroupla referencia de la API de Amazon Logs.

Console

Especifique el grupo de registros de imágenes o el grupo de registros de canalización para su canalización en la sección Configuración de registros, en Configuración avanzada.

CLI

Especifica los siguientes campos dentro del `logging-configuration` objeto si utilizas un objeto JSON para la configuración:

- `imageLogGroupName`
- `pipelineLogGroupName`

Para especificar todos los parámetros directamente en la línea de comandos, consulte create-image-pipelinela Referencia de AWS CLI comandos.

Image Builder utiliza los siguientes grupos de registros predeterminados si no se especifica un grupo de registro personalizado:

Registros de creación de imágenes

Image Builder escribe los registros de compilación en el siguiente grupo y flujo de CloudWatch registros de Image Builder:

LogGroup: `/aws/imagebuilder/`*`ImageName`*

LogStream (x.x.x/x): *`ImageVersion/ImageBuildVersion`*

Registros de ejecución de canalización

Image Builder escribe los registros de ejecución de la canalización en el siguiente grupo y flujo de CloudWatch registros de Image Builder:

LogGroup: `/aws/imagebuilder/pipeline/`*`pipeline-name`*

LogStream: *`2025/09/01`* (la fecha de ejecución de la canalización en YYYY/MM/DD formato)

Cada registro de canalización se adjunta a la transmisión de ese día.

Ejecute una canalización de imágenes manualmente

Si ha elegido la opción de programación manual para la canalización, solo se ejecutará cuando inicie la compilación manualmente. Si ha elegido una de las opciones de programación automática, también puede ejecutarla manualmente, entre las ejecuciones programadas de forma regular.

Por ejemplo, si tiene una canalización que normalmente se ejecuta una vez al mes, pero necesita incorporar una actualización a uno de sus componentes dos semanas después de la ejecución anterior, puede optar por ejecutar la canalización manualmente.

Console

Para ejecutar la canalización desde la página de detalles de la canalización de la consola de Generador de Imágenes, seleccione Ejecutar canalización en el menú Acciones de la parte superior de la página. Aparece un mensaje de estado en la parte superior de la página para notificarle que su canalización se ha iniciado o si se ha producido un error.

1. En la esquina superior izquierda de la página de detalles de la canalización, seleccione Ejecutar canalización en el menú Acciones.
2. Puede ver el estado actual de su canalización en la pestaña Imágenes de salida, en la columna Estado.

AWS CLI

En el siguiente ejemplo se muestra cómo utilizar el comando start-image-pipeline-execution para iniciar una canalización de imágenes en la AWS CLI de forma manual. Al ejecutar este comando, la canalización crea y distribuye una imagen nueva.

```
aws imagebuilder start-image-pipeline-execution --image-pipeline-arn
 arn:aws:imagebuilder:us-west-2:111122223333:image-pipeline/my-example-pipeline
```

Para ver qué recursos se crean cuando se ejecuta la canalización de compilación, consulte Recursos creados.

Utilizar expresiones cron en Generador de imágenes

Usa expresiones cron para EC2 Image Builder a fin de configurar un intervalo de tiempo para actualizar tu imagen con actualizaciones que se apliquen a la imagen base y a los componentes de tu canalización. El intervalo de tiempo para la actualización de la canalización comienza con la hora que establezca en la expresión cron. Puede establecer la hora en su expresión cron hasta el minuto. La compilación de canalización puede ejecutarse a la hora de inicio o después.

A veces, la compilación puede tardar unos segundos o hasta un minuto en comenzar a ejecutarse.

> **Note**
>
> Las expresiones cron utilizan la hora universal coordinada (UTC) de forma predeteminada, o se puede especificar la zona horaria. Para obtener más información sobre la hora UTC y encontrar el desfase de su zona horaria, consulte Abreviaturas de zonas horarias: lista mundial.

Valores admitidos para expresiones cron en Generador de Imágenes

EC2 Image Builder utiliza un formato cron que consta de seis campos obligatorios. Cada uno está separado de los demás por un espacio intermedio, sin espacios iniciales ni finales:

```
<Minute> <Hour> <Day> <Month> <Day of the week> <Year>
```

En la siguiente tabla se desglosan los valores compatibles para las entradas cron necesarias.

Valores admitidos para expresiones cron

Campo	Valores	Caracteres comodín
Minuto	`0-59`	`, - * /`
Hora	`0-23`	`, - * /`
Día	`1-31`	`, - * ? / L W`
Mes	`1-12` o `jan-dec`	`, - * /`
Día de la semana	`1-7` o `sun-sat`	`, - * ? L #`
Año	`1970-2199`	`, - * /`

Caracteres comodín

En la siguiente tabla se describe cómo Generador de Imágenes utiliza los caracteres comodín para las expresiones cron. Tenga en cuenta que la compilación puede tardar hasta un minuto después de la hora que especifique para que comience la compilación.

Comodines admitidos para expresiones cron

Comodín	Descripción
,	El carácter comodín , (coma) incluye valores adicionales. En el campo Mes, `jan,feb,mar` incluye enero, febrero y marzo.
-	El carácter comodín - (guion) especifica los intervalos. En el campo día del mes, `1-15` incluye los días del 1 al 15 del mes especific ado.
*	El carácter comodín * (asterisco) incluye todos los valores válidos para el campo.
?	El carácter comodín ? (signo de interrogación) especifica que el valor del campo depende de otra configuración. En el caso del día y Day-of-week los campos, si se especifica uno o incluye todos los valores posibles (*), el otro debe ser un?. No puede especificar ambos. Por ejemplo, si introduce a 7 en el campo Día (ejecuta la compilación el séptimo día del mes), la Day-of-week posición debe contener un?.
/	El comodín / (barra inclinada) especifica incrementos. Por ejemplo, si desea que la compilación se ejecute cada dos días, escriba `*/2` en el campo día.
L	El carácter comodín L en cualquiera de los campos de día especifica el último día: del 28 al 31 para el día del mes, según el mes, o el domingo, para el día de la semana.
W	El comodín W en el campo Day-of-month especifica un día de la semana. En el Day-of-month campo, si escribes un número antes

Comodín	Descripción
	delW, significa que quieres enfocarte en el día de la semana más cercano a ese día. Por ejemplo, si especifica 3W, quiere que la compilación se ejecute el día de la semana más cercano al tercer día del mes.
#	El símbolo # (almohadilla) solo está permitido para el campo del día de la semana y debe ir seguido de un número entre 1 y 5. El número especifica qué semanas de un mes determina do se utilizan para que se ejecute la compilaci ón. Por ejemplo, si quiere que la compilaci ón se ejecute el segundo viernes de cada mes, utilice fri#2 para el campo del día de la semana.

Restricciones

- No se pueden especificar los campos Day-of-month y Day-of-week en la misma expresión Cron. Si especifica un valor o * en uno de estos campos, debe utilizar un ? en el otro.
- No se admiten expresiones cron que produzcan frecuencias superiores a un minuto.

Ejemplos de expresiones cron en Generador de imágenes

Las expresiones cron se escriben de forma diferente para la consola de Generador de Imágenes que para la API o la CLI. Para ver ejemplos, elija la pestaña que corresponda a su caso.

Image Builder console

Los siguientes ejemplos muestran expresiones cron que puede introducir en la consola para su programa de compilación. La hora UTC se especifica usando un formato de 24 horas.

Ejecutar a diario a las 10:00 (UTC)

```
0 10 * * ? *
```

Ejecutar a diario a las 12:15 (UTC)

```
15 12 * * ? *
```

Ejecutar a diario a la medianoche (UTC)

```
0 0 * * ? *
```

Ejecución a las 10:00 (UTC) de lunes a viernes por la mañana

```
0 10 ? * 2-6 *
```

Ejecución a las 18:00 (UTC) de lunes a viernes por la noche

```
0 18 ? * mon-fri *
```

Ejecución a las 08:00 (UTC) el primer día de cada mes

```
0 8 1 * ? *
```

Ejecutar el segundo martes de cada mes a las 22:30 (UTC)

```
30 22 ? * tue#2 *
```

> **Tip**
>
> Si no quiere que su trabajo de canalización se prolongue hasta el día siguiente mientras esté en ejecución, asegúrese de tener en cuenta la hora de compilación cuando especifique la hora de inicio.

API/CLI

Los siguientes ejemplos muestran expresiones cron que puede ingresar para su programa de compilación usando comandos CLI o solicitudes de API. Solo se muestra la expresión cron.

Ejecutar a diario a las 10:00 (UTC)

```
cron(0 10 * * ? *)
```

Ejecutar a diario a las 12:15 (UTC)

```
cron(15 12 * * ? *)
```

Ejecutar a diario a la medianoche (UTC)

```
cron(0 0 * * ? *)
```

Ejecución a las 10:00 (UTC) de lunes a viernes por la mañana

```
cron(0 10 ? * 2-6 *)
```

Ejecución a las 18:00 (UTC) de lunes a viernes por la noche

```
cron(0 18 ? * mon-fri *)
```

Ejecución a las 08:00 (UTC) el primer día de cada mes

```
cron(0 8 1 * ? *)
```

Ejecutar el segundo martes de cada mes a las 22:30 (UTC)

```
cron(30 22 ? * tue#2 *)
```

> **Tip**
>
> Si no quiere que su trabajo de canalización se prolongue hasta el día siguiente mientras esté en ejecución, asegúrese de tener en cuenta la hora de compilación cuando especifique la hora de inicio.

Expresiones de frecuencia en Generador de imágenes

Una expresión de frecuencia comienza cuando se crea una regla de evento programado y, a continuación, se ejecuta en su programa definido.

Las expresiones de frecuencia tienen dos campos obligatorios. Los campos están separados por un espacio en blanco.

Sintaxis

```
rate(value unit)
```

valor

Un número positivo.

unidad

La unidad de tiempo. Se requieren diferentes unidades para valores de 1, como `minute`, y valores superiores a 1, como `minutes`.

Valores válidos: minuto | minutos | hora | horas | día | días

Restricciones

Si el valor es igual a 1, entonces la unidad debe ser singular. Del mismo modo, para valores mayores que 1, la unidad debe ser plural. Por ejemplo, `rate(1 hours)` y `rate(5 hour)` no son válidos, pero `rate(1 hour)` y `rate(5 hours)` son válidos.

Enumeración y visualización de los detalles de la canalización

En esta sección se describen las distintas formas de encontrar información y ver los detalles de las canalizaciones de imágenes de EC2 Image Builder.

Detalles de la canalización

- Enumere las canalizaciones de imágenes de AWS CLI
- Obtenga los detalles de la canalización de imágenes en la AWS CLI

Enumere las canalizaciones de imágenes de AWS CLI

En el siguiente ejemplo, se muestra cómo utilizar el list-image-pipelines comando AWS CLI para enumerar todas las canalizaciones de imágenes.

```
aws imagebuilder list-image-pipelines
```

Obtenga los detalles de la canalización de imágenes en la AWS CLI

En el siguiente ejemplo, se muestra cómo utilizar el get-image-pipeline comando de AWS CLI para obtener los detalles de una canalización de imágenes a través de su ARN.

```
aws imagebuilder get-image-pipeline --image-pipeline-arn arn:aws:imagebuilder:us-
west-2:123456789012:image-pipeline/my-example-pipeline
```

Creación y actualización de canalizaciones de imágenes de AMI

Puede configurar, gestionar y gestionar las canalizaciones de imágenes de AMI desde la consola Image Builder, mediante la API Image Builder SDKs o la AWS CLI. En la consola, puede utilizar el asistente de consola Crear canalización de imágenes para guiarlo en los siguientes pasos.

- Especifique los detalles de la canalización, como el nombre, la descripción y las etiquetas de recursos.
- Configure el cronograma de canalización y los valores predeterminados de registro. Para la ejecución programada de la canalización, puede establecer el número de errores consecutivos que se permiten antes de que Image Builder desactive la canalización.
- Seleccione una receta de imagen de AMI que incluya una imagen base de imágenes gestionadas por Amazon de inicio rápido, imágenes que haya creado o que haya compartido con usted o imágenes a las que se suscriba a través de. AWS Marketplace La receta también incluye componentes que realizan las siguientes tareas en las EC2 instancias que Image Builder utiliza para crear la imagen:
  - Cómo agregar y eliminar software.
  - Personalización de los ajustes y los scripts
  - Ejecución de las pruebas seleccionadas
- Especifique los flujos de trabajo para configurar los pasos de creación y prueba de imágenes que ejecuta la canalización.
- Defina la configuración de infraestructura para la canalización con las opciones predeterminadas o las opciones que configure usted. La configuración incluye el tipo de instancia y el par de claves que se utilizarán para la imagen, las opciones de seguridad y red, las opciones de almacenamiento de registros y solución de problemas y las notificaciones de SNS.

  Se trata de un paso opcional. Generador de imágenes utiliza las opciones predeterminadas para la configuración de la infraestructura si no define la configuración usted.
- Defina las opciones de distribución para entregar las imágenes a las cuentas y regiones de AWS de destino. Puede especificar una clave de KMS para el cifrado, configurar el uso compartido de AMI o la configuración de licencias, o configurar una plantilla de lanzamiento para la AMIs que distribuya.

  Se trata de un paso opcional. Si no define la configuración usted, Generador de imágenes utilizará un nombre predeterminado para la AMI de salida y distribuirá la AMI a la región de origen. La región de origen es la región donde se ejecuta la canalización.

Para obtener más información y un step-by-step tutorial sobre el uso del asistente de consola Create Image Pipeline con los valores predeterminados, si está disponible, consulteTutorial: creación de una canalización de imágenes con una AMI de salida desde el asistente de la consola de Generador de Imágenes.

Contenido

- Cree una canalización de imágenes AMI desde AWS CLI
- Actualización de las canalizaciones de imágenes de AMI desde la consola
- Actualice las canalizaciones de imágenes de AMI desde AWS CLI

Cree una canalización de imágenes AMI desde AWS CLI

Para crear una canalización de imágenes a partir de AWS CLI, ejecute el create-image-pipeline comando con las opciones de configuración aplicables a la canalización. Tienes la opción de crear un archivo JSON que contenga toda la configuración de la canalización o de especificar la configuración en tiempo de ejecución. En esta sección, se utiliza el método del archivo de configuración JSON para simplificar el comando.

La frecuencia con la que la canalización crea una nueva imagen para incorporar las actualizaciones pendientes de la imagen base y los componentes depende de la `schedule` que haya configurado. Una `schedule` tiene los siguientes atributos:

- `scheduleExpression`: establece el cronograma de ejecución de la canalización para evaluar `pipelineExecutionStartCondition` y determinar si se debe iniciar una compilación. El cronograma se configura con expresiones cron. Para obtener más información sobre cómo dar formato a una expresión cron en Generador de Imágenes, consulte Utilizar expresiones cron en Generador de imágenes.
- `pipelineExecutionStartCondition`: determina si su canalización debe iniciar la compilación. Los valores válidos son:
  - `EXPRESSION_MATCH_ONLY`: la canalización creará una nueva imagen cada vez que la expresión cron coincida con la hora actual.
  - `EXPRESSION_MATCH_AND_DEPENDENCY_UPDATES_AVAILABLE`: la canalización no iniciará la compilación de una nueva imagen a menos que haya cambios pendientes en su imagen base o en los componentes.

Al ejecutar el create-image-pipeline comando en el AWS CLI, muchos de los recursos de configuración son opcionales. Sin embargo, algunos de los recursos tienen requisitos condicionales, según el tipo de imagen que crea la canalización. Los siguientes identificadores de recursos son necesarios para las canalizaciones de imágenes de AMI:

- El ARN de la receta de la imagen
- Configuración de infraestructura ARN

Ejemplo: crear una imagen de Windows 2019

En este ejemplo, se configura una canalización que está programada para ejecutarse una vez a la semana los domingos. El archivo de configuración que se muestra en el primer paso utiliza los recursos existentes para la configuración de la imagen, la infraestructura y la distribución, junto con otros ajustes, para crear una imagen de Windows 2019.

1. Cree un archivo de configuración (opcional)

   En este ejemplo, se utiliza un archivo de configuración denominado `create-image-pipeline.json` para configurar los ajustes en un solo lugar. Como alternativa, puede usar las opciones de la línea de comandos al ejecutar el comando para especificar todos los detalles que se muestran aquí en el archivo de configuración.

```
{
 "name": "ExampleWindows2019Pipeline",
 "description": "Builds Windows 2019 Images",
 "enhancedImageMetadataEnabled": true,
 "imageRecipeArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/my-example-recipe/2020.12.03",
 "infrastructureConfigurationArn": "arn:aws:imagebuilder:us-west-2:123456789012:infrastructure-configuration/my-example-infrastructure-configuration",
 "distributionConfigurationArn": "arn:aws:imagebuilder:us-west-2:123456789012:distribution-configuration/my-example-distribution-configuration",
 "imageTestsConfiguration": {
  "imageTestsEnabled": true,
  "timeoutMinutes": 60
 },
 "schedule": {
```

```
  "scheduleExpression": "cron(0 0 * * SUN *)",
  "pipelineExecutionStartCondition":
 "EXPRESSION_MATCH_AND_DEPENDENCY_UPDATES_AVAILABLE"
 },
 "status": "ENABLED"
}
```

> **Note**
>
> - Debe incluir la notación `file://` al principio de la ruta del archivo JSON.
> - La ruta del archivo JSON debe seguir la convención apropiada para el sistema operativo base donde se está ejecutando el comando. Por ejemplo, Windows utiliza la barra diagonal inversa (\) para referirse a la ruta de directorios, mientras que Linux y macOS usan la barra diagonal (/).

2. Ejecute el comando para crear una imagen

   En este ejemplo, se utiliza el archivo de configuración creado en el primer paso como entrada del `create-image-pipeline` comando. Como alternativa, puedes especificar la configuración y los recursos de tu canalización directamente al ejecutar el comando. Para obtener más información consulte create-image-pipeline en la Referencia de la AWS CLI .

   ```
   aws imagebuilder create-image-pipeline --cli-input-json file://create-image-
   pipeline.json
   ```

Actualización de las canalizaciones de imágenes de AMI desde la consola

Tras crear una canalización de imágenes de Generador de Imágenes para la imagen de AMI, puede realizar cambios en la configuración de la infraestructura y en los ajustes de distribución desde la consola de Generador de Imágenes.

Para actualizar una canalización de imágenes con una nueva receta de imágenes, debe utilizar la AWS CLI. Para obtener más información, consulte la sección Actualice las canalizaciones de imágenes de AMI desde AWS CLI de esta guía.

Elección de una canalización de Generador de Imágenes existente

1. Abra la consola EC2 de Image Builder en https://console.aws.amazon.com/imagebuilder/.

2. Para ver una lista de las canalizaciones de imágenes creadas en su cuenta, elija canalizaciones de imágenes en el panel de navegación.

> **Note**
>
> La lista de canalizaciones de imágenes incluye un indicador del tipo de imagen de salida que crea la canalización: AMI o Docker.

3. Para ver los detalles o editar una canalización, elija el enlace con el nombre de la canalización. Esto abre la vista detallada de la canalización.

> **Note**
>
> También puede seleccionar la casilla junto al nombre de la canalización, y, a continuación, Ver detalles.

Detalles de la canalización

La página Detalles de la canalización incluye las siguientes secciones:

Resumen

La sección en la parte superior de la página resume los detalles clave de la canalización que están visibles con cualquiera de las pestañas de detalles abiertas. Los detalles que se muestran en esta sección solo se pueden editar en sus respectivas pestañas de detalles.

Pestaña de Detalles

- Imágenes de salida: muestra las imágenes de salida que ha producido la canalización.
- Receta de imagen: muestra los detalles de la receta. Después de crear una receta, no la puede editar. Debe crear una nueva versión de la receta desde la página Recetas de imágenes de la consola de Generador de Imágenes o mediante los comandos de Generador de Imágenes en la AWS CLI. Para obtener más información, consulte Administración de recetas en Generador de imágenes.
- Configuración de infraestructura: muestra información editable para configurar la infraestructura de canalización de compilación.
- Ajustes de distribución: muestra información editable para la distribución de la AMI.

- EventBridge reglas: para el bus de eventos seleccionado, muestra EventBridge las reglas que se dirigen a la canalización actual. Incluye las acciones Crear bus de eventos y Crear reglas que enlazan con la EventBridge consola. Para obtener más información acerca de esta pestaña, consulte Utilice reglas EventBridge .

Edición de la configuración de la infraestructura de su canalización

La configuración de la infraestructura incluye los siguientes detalles que puede editar después de crear la canalización:

- La descripción de la configuración de la infraestructura.
- El rol de IAM que asociará con el perfil de instancia.
- AWS infraestructura, incluido el tipo de instancia y un tema de SNS para las notificaciones.
- VPC, la subred y los grupos de seguridad.
- La configuración de solución de problemas, incluida la finalización de la instancia en caso de falla, el par de claves para la conexión y una ubicación de bucket de S3 opcional para los registros de las instancias.

Para editar la configuración de la infraestructura desde la página de detalles de canalización, siga estos pasos:

1. Seleccione la pestaña Configuración de infraestructura.
2. Seleccione Editar en la esquina superior derecha del panel Detalles de configuración.
3. Cuando esté listo para guardar las actualizaciones que ha realizado en su configuración de la infraestructura, seleccione Guardar cambios.

Edición de los ajustes de distribución de su canalización

La configuración de distribución incluye los siguientes detalles que puede editar después de crear la canalización:

- La descripción de la configuración de su distribución.
- Ajustes de región para las regiones donde distribuye la imagen. La región 1 establece de forma predeterminada la región en la que creó la canalización. Puede añadir regiones para su distribución con el botón Añadir región y eliminar todas las regiones, excepto la región 1.

Los ajustes de región incluyen:

- Región de destino.
- El nombre de la AMI de salida
- Lanzamiento de los permisos y las cuentas con las que compartirlos
- Licencias asociadas (configuraciones de licencias asociadas)

> **Note**
>
> La configuración del License Manager no se replicará en AWS las regiones que deban estar habilitadas en su cuenta, por ejemplo, entre las regiones `ap-east-1` (Hong Kong) y `me-south-1` (Baréin).

Para editar los ajustes de distribución desde la página de detalles de la canalización, siga estos pasos:

1. Seleccione la pestaña ajustes de distribución.
2. Seleccione Editar en la esquina superior derecha del panel Detalles de distribución.
3. Cuando esté listo para guardar las actualizaciones, seleccione Guardar cambios.

Edición del cronograma de creación de su canalización

La página Editar canalización incluye los siguientes detalles que puede editar después de crear la canalización:

- La descripción de su canalización.
- Recopilación mejorada de metadatos. Esto está activado de forma predeterminada. Para desactivarla, desactive la casilla Habilitar la recopilación mejorada de metadatos.
- El cronograma de compilación de su canalización. Puede cambiar las opciones de programación y todos los ajustes aquí.

Para editar su canalización desde la página de detalles de la canalización, siga estos pasos:

1. En la esquina superior derecha de la página de detalles de la canalización, seleccione Acciones y, a continuación, Editar canalización.

2. Cuando esté listo para guardar las actualizaciones, seleccione Guardar cambios.

> **Note**
>
> Para obtener más información sobre cómo programar su compilación usando expresiones Cron, consulte Utilizar expresiones cron en Generador de imágenes.

Actualice las canalizaciones de imágenes de AMI desde AWS CLI

Puede actualizar una canalización de imágenes de AMI mediante un archivo JSON como entrada al comando update-image-pipeline en la AWS CLI. Para configurar el archivo JSON, debe tener Amazon Resource Names (ARNs) para hacer referencia a los siguientes recursos existentes:

- Canalización de imágenes que se va a actualizar
- Receta de imagen
- Configuración de infraestructura
- Ajustes de la distribución

Puede actualizar una canalización de imágenes de AMI con el AWS CLI siguiente update-image-pipeline comando:

> **Note**
>
> UpdateImagePipeline no admite actualizaciones selectivas para la canalización. Debe especificar todas las propiedades obligatorias en la solicitud de actualización, no solo las propiedades que han cambiado.

1. Creación de un archivo JSON de entrada de la CLI

   Utilice su herramienta de edición de archivos favorita para crear un archivo JSON con las siguientes claves, además de valores que sean válidos para su entorno. En este ejemplo, se utiliza un archivo con el nombre `create-component.json`:

   ```
   {
   ```

```
 "imagePipelineArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-
pipeline/my-example-pipeline",
 "imageRecipeArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/my-
example-recipe/2019.12.08",
 "infrastructureConfigurationArn": "arn:aws:imagebuilder:us-
west-2:123456789012:infrastructure-configuration/my-example-infrastructure-
configuration",
 "distributionConfigurationArn": "arn:aws:imagebuilder:us-
west-2:123456789012:distribution-configuration/my-example-distribution-
configuration",
 "imageTestsConfiguration": {
  "imageTestsEnabled": true,
  "timeoutMinutes": 120
 },
 "schedule": {
  "scheduleExpression": "cron(0 0 * * MON *)",
  "pipelineExecutionStartCondition":
 "EXPRESSION_MATCH_AND_DEPENDENCY_UPDATES_AVAILABLE"
 },
 "status": "DISABLED"
}
```

> **Note**
>
> - Debe incluirla notación `file://` al principio de la ruta del archivo JSON.
> - La ruta del archivo JSON debe seguir la convención apropiada para el sistema operativo base donde se está ejecutando el comando. Por ejemplo, Windows utiliza la barra diagonal inversa (\) para referirse a la ruta de directorios, mientras que Linux y macOS usan la barra diagonal (/).

2. Ejecute el siguiente comando utilizando el archivo que creó como entrada.

```
aws imagebuilder update-image-pipeline --cli-input-json file://update-image-
pipeline.json
```

Creación y actualización de canalizaciones de imágenes de contenedor

Puede establecer, configurar y administrar las canalizaciones de imágenes de contenedor mediante la consola Generador de Imágenes, a través de la API de Generador de Imágenes o con los comandos imagebuilder en la AWS CLI. El asistente de consola Crear canalizaciones de imágenes proporciona artefactos de inicio y lo guía paso a paso para:

- Selección de una imagen base de los repositorios de imágenes administradas de inicio rápido, Amazon ECR o Docker Hub
- Cómo agregar y eliminar software.
- Personalización de los ajustes y los scripts
- Ejecución de las pruebas seleccionadas
- Crear un Dockerfile con variables de tiempo de compilación preconfiguradas.
- Distribuya imágenes a las regiones AWS

Para obtener más información y un step-by-step tutorial sobre el uso del asistente de consola Create Image Pipeline, consulteTutorial: creación de una canalización de imágenes con una imagen de contenedor de Docker de salida desde el asistente de consola de Generador de imágenes.

Contenido

- Cree una canalización de imágenes de contenedores a partir del AWS CLI
- Actualización de una canalización de imágenes de contenedor desde la consola
- Actualice las canalizaciones de imágenes de contenedores desde AWS CLI

Cree una canalización de imágenes de contenedores a partir del AWS CLI

Para crear una canalización de imágenes a partir de AWS CLI, ejecute el create-image-pipeline comando con las opciones de configuración aplicables a la canalización. Tienes la opción de crear un archivo JSON que contenga toda la configuración de la canalización o de especificar la configuración en tiempo de ejecución. En esta sección, se utiliza el método del archivo de configuración JSON para simplificar el comando.

La frecuencia con la que la canalización crea una nueva imagen para incorporar las actualizaciones pendientes de la imagen base y los componentes depende de la `schedule` que haya configurado. Una `schedule` tiene los siguientes atributos:

- `scheduleExpression`: establece el cronograma de ejecución de la canalización para evaluar `pipelineExecutionStartCondition` y determinar si se debe iniciar una compilación. El cronograma se configura con expresiones cron. Para obtener más información sobre cómo dar formato a una expresión cron en Generador de Imágenes, consulte Utilizar expresiones cron en Generador de imágenes.
- `pipelineExecutionStartCondition`: determina si su canalización debe iniciar la compilación. Los valores válidos son:
  - `EXPRESSION_MATCH_ONLY`: la canalización creará una nueva imagen cada vez que la expresión cron coincida con la hora actual.
  - `EXPRESSION_MATCH_AND_DEPENDENCY_UPDATES_AVAILABLE`: la canalización no iniciará la compilación de una nueva imagen a menos que haya cambios pendientes en su imagen base o en los componentes.

Al ejecutar el create-image-pipeline comando en el AWS CLI, muchos de los recursos de configuración son opcionales. Sin embargo, algunos de los recursos tienen requisitos condicionales, según el tipo de imagen que crea la canalización. Las canalizaciones de imágenes de contenedor requieren los siguientes recursos:

- ARN de receta de contenedor
- Configuración de infraestructura ARN

Si no incluye un recurso de configuración de distribución al ejecutar el comando create-image-pipeline, la imagen de salida se almacena en el repositorio de ECR que especifique como repositorio de destino en su receta de contenedor en la región donde se ejecuta el comando. Si incluye un recurso de configuración de distribución para la canalización, se utilizará el repositorio de destino que se haya especificado para la primera región de la distribución.

1. Creación de un archivo JSON de entrada de la CLI

   Utilice su herramienta de edición de archivos favorita para crear un archivo JSON con las siguientes claves, además de valores que sean válidos para su entorno. En este ejemplo, se utiliza un archivo con el nombre `create-image-pipeline.json`:

```
{
 "name": "MyWindows2019Pipeline",
 "description": "Builds Windows 2019 Images",
 "enhancedImageMetadataEnabled": true,
 "containerRecipeArn": "arn:aws:imagebuilder:us-west-2:123456789012:container-
recipe/my-example-recipe/2020.12.03",
 "infrastructureConfigurationArn": "arn:aws:imagebuilder:us-
west-2:123456789012:infrastructure-configuration/my-example-infrastructure-
configuration",
 "distributionConfigurationArn": "arn:aws:imagebuilder:us-
west-2:123456789012:distribution-configuration/my-example-distribution-
configuration",
 "imageTestsConfiguration": {
  "imageTestsEnabled": true,
  "timeoutMinutes": 60
 },
 "schedule": {
  "scheduleExpression": "cron(0 0 * * SUN *)",
  "pipelineExecutionStartCondition":
 "EXPRESSION_MATCH_AND_DEPENDENCY_UPDATES_AVAILABLE"
 },
 "status": "ENABLED"
}
```

> **Note**
>
> - Debe incluirla notación `file://` al principio de la ruta del archivo JSON.
> - La ruta del archivo JSON debe seguir la convención apropiada para el sistema operativo base donde se está ejecutando el comando. Por ejemplo, Windows utiliza la barra diagonal inversa (\) para referirse a la ruta de directorios, mientras que Linux y macOS usan la barra diagonal (/).

2. Ejecute el siguiente comando utilizando el archivo que creó como entrada.

```
aws imagebuilder create-image-pipeline --cli-input-json file://create-image-
pipeline.json
```

Actualización de una canalización de imágenes de contenedor desde la consola

Después de crear una canalización de imágenes de contenedor de Generador de Imágenes para la imagen de Docker, puede realizar cambios en la configuración de la infraestructura y de distribución desde la consola de Generador de Imágenes.

Para actualizar una canalización de imágenes de contenedor con una nueva receta de contenedor, debe usar la AWS CLI. Para obtener más información, consulte la sección Actualice las canalizaciones de imágenes de contenedores desde AWS CLI de esta guía.

Elija una canalización de imagen de Docker de Generador de Imágenes existente.

1. Abra la consola EC2 de Image Builder en https://console.aws.amazon.com/imagebuilder/.
2. Para ver una lista de las canalizaciones de imágenes creadas en su cuenta, elija canalizaciones de imágenes en el panel de navegación.

   > **Note**
   >
   > La lista de canalizaciones de imágenes incluye un indicador del tipo de imagen de salida que crea la canalización: AMI o Docker.

3. Para ver los detalles o editar una canalización, elija el enlace con el nombre de la canalización. Esto abre la vista detallada de la canalización.

   > **Note**
   >
   > También puede seleccionar la casilla junto al nombre de la canalización, y, a continuación, Ver detalles.

Detalles de la canalización

La página de detalles de la canalización de EC2 Image Builder incluye las siguientes secciones:

Resumen

La sección en la parte superior de la página resume los detalles clave de la canalización que están visibles con cualquiera de las pestañas de detalles abiertas. Los detalles que se muestran en esta sección solo se pueden editar en sus respectivas pestañas de detalles.

Pestaña de Detalles

- Imágenes de salida: muestra las imágenes de salida que ha producido la canalización.
- Receta de contenedor: muestra los detalles de la receta. Después de crear una receta, no la puede editar. Debe crear una nueva versión de la receta desde la página Recetas de contenedor. Para obtener más información, consulte Creación de una nueva versión de una receta de contenedor.
- Configuración de infraestructura: muestra información editable para configurar la infraestructura de canalización de compilación.
- Ajustes de distribución: muestra información editable para la distribución de imagen de Docker.
- EventBridge reglas: para el bus de eventos seleccionado, muestra EventBridge las reglas que se dirigen a la canalización actual. Incluye las acciones Crear bus de eventos y Crear reglas que enlazan con la EventBridge consola. Para obtener más información acerca de esta pestaña, consulte Utilice reglas EventBridge .

Edición de la configuración de la infraestructura de su canalización

La configuración de la infraestructura incluye los siguientes detalles que puede editar después de crear la canalización:

- La descripción de la configuración de la infraestructura.
- El rol de IAM que asociará con el perfil de instancia.
- AWS infraestructura, incluido el tipo de instancia y un tema de SNS para las notificaciones.
- VPC, la subred y los grupos de seguridad.
- La configuración de solución de problemas, incluida la finalización de la instancia en caso de falla, el par de claves para la conexión y una ubicación de bucket de S3 opcional para los registros de las instancias.

Para editar la configuración de la infraestructura desde la página de detalles de canalización, siga estos pasos:

1. Seleccione la pestaña Configuración de infraestructura.

2. Seleccione Editar en la esquina superior derecha del panel Detalles de configuración.
3. Cuando esté listo para guardar las actualizaciones que ha realizado en su configuración de la infraestructura, seleccione Guardar cambios.

Edición de los ajustes de distribución de su canalización

La configuración de distribución incluye los siguientes detalles que puede editar después de crear la canalización:

- La descripción de sus ajustes de distribución.
- Ajustes de región para las regiones donde distribuye la imagen. La región 1 establece de forma predeterminada la región en la que creó la canalización. Puede añadir regiones para su distribución con el botón Añadir región y eliminar todas las regiones, excepto la región 1.

  Los ajustes de región incluyen:

  - Región de destino.
  - El servicio tiene el valor predeterminado “ECR” y no se puede editar.
  - Nombre del repositorio: el nombre del repositorio de destino (no incluye la ubicación de Amazon ECR). Por ejemplo, el nombre del repositorio con la ubicación tendría el siguiente patrón:

    `<account-id>.dkr.ecr.<region>.amazonaws.com/<repository-name>`

    > **Note**
    >
    > Si cambia el nombre del repositorio, solo las imágenes creadas después del cambio de nombre se añadirán al nuevo nombre. Todas las imágenes anteriores que haya creado su canalización permanecen en su repositorio original.

Para editar los ajustes de distribución desde la página de detalles de la canalización, siga estos pasos:

1. Seleccione la pestaña ajustes de distribución.
2. Seleccione Editar en la esquina superior derecha del panel Detalles de distribución.
3. Cuando esté listo para guardar las actualizaciones que haya realizado en los ajustes de distribución, seleccione Guardar cambios.

Edición del cronograma de creación de su canalización

La página Editar canalización incluye los siguientes detalles que puede editar después de crear la canalización:

- La descripción de su canalización.
- Recopilación mejorada de metadatos. Esto está activado de forma predeterminada. Para desactivarla, desactive la casilla Habilitar la recopilación mejorada de metadatos.
- El cronograma de compilación de su canalización. Puede cambiar las opciones de cronograma y todas las configuraciones de esta sección.

Para editar su canalización desde la página de detalles de la canalización, siga estos pasos:

1. En la esquina superior derecha de la página de detalles de la canalización, seleccione Acciones y, a continuación, Editar canalización.
2. Cuando esté listo para guardar las actualizaciones, seleccione Guardar cambios.

> **Note**
>
> Para obtener más información sobre cómo programar su compilación usando expresiones Cron, consulte Utilizar expresiones cron en Generador de imágenes.

Actualice las canalizaciones de imágenes de contenedores desde AWS CLI

Puede actualizar una canalización de imágenes de contenedor utilizando un archivo JSON como entrada para el comando update-image-pipeline en la AWS CLI. Para configurar el archivo JSON, debe tener Amazon Resource Names (ARNs) para hacer referencia a los siguientes recursos existentes:

- Canalización de imágenes que se va a actualizar
- Receta de contenedor
- Configuración de infraestructura
- Ajustes de distribución (si está incluida en la canalización actual)

> **Note**
>
> Si se incluye el recurso de configuración de distribución, el repositorio de ECR que está especificado como repositorio de destino en los ajustes de distribución de la región donde se ejecuta el comando (región 1) tiene prioridad sobre el repositorio de destino especificado en la receta de contenedor.

Siga estos pasos para actualizar una canalización de imágenes de contenedor mediante el comando update-image-pipeline en la AWS CLI:

> **Note**
>
> UpdateImagePipeline no admite actualizaciones selectivas para la canalización. Debe especificar todas las propiedades obligatorias en la solicitud de actualización, no solo las propiedades que han cambiado.

1. Creación de un archivo JSON de entrada de la CLI

   Utilice su herramienta de edición de archivos favorita para crear un archivo JSON con las siguientes claves, además de valores que sean válidos para su entorno. En este ejemplo, se utiliza un archivo con el nombre `create-component.json`:

```
{
 "imagePipelineArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-
pipeline/my-example-pipeline",
 "containerRecipeArn": "arn:aws:imagebuilder:us-west-2:123456789012:container-
recipe/my-example-recipe/2020.12.08",
 "infrastructureConfigurationArn": "arn:aws:imagebuilder:us-
west-2:123456789012:infrastructure-configuration/my-example-infrastructure-
configuration",
 "distributionConfigurationArn": "arn:aws:imagebuilder:us-
west-2:123456789012:distribution-configuration/my-example-distribution-
configuration",
 "imageTestsConfiguration": {
  "imageTestsEnabled": true,
  "timeoutMinutes": 120
 },
 "schedule": {
```

```
  "scheduleExpression": "cron(0 0 * * MON *)",
  "pipelineExecutionStartCondition":
 "EXPRESSION_MATCH_AND_DEPENDENCY_UPDATES_AVAILABLE"
 },
 "status": "DISABLED"
}
```

> **Note**
>
> - Debe incluirla notación `file://` al principio de la ruta del archivo JSON.
> - La ruta del archivo JSON debe seguir la convención apropiada para el sistema operativo base donde se está ejecutando el comando. Por ejemplo, Windows utiliza la barra diagonal inversa (\) para referirse a la ruta de directorios, mientras que Linux y macOS usan la barra diagonal (/).

2. Ejecute el siguiente comando utilizando el archivo que creó como entrada.

```
aws imagebuilder update-image-pipeline --cli-input-json file://update-image-
pipeline.json
```

Configuración de flujos de trabajo de canalizaciones de imágenes en Generador de imágenes

Con los flujos de trabajo de imágenes, puede personalizar los flujos de trabajo que ejecuta una canalización para crear y probar imágenes según sus necesidades. Los flujos de trabajo que defina se ejecutan en el contexto del marco del flujo de trabajo de Generador de imágenes. Para obtener más información sobre las etapas que componen el marco del flujo de trabajo, consulte Administración de flujos de trabajo de creación y prueba para imágenes de Generador de imágenes.

Flujo de trabajo de creación

Los flujos de trabajo de creación se ejecutan durante la etapa `Build` del marco del flujo de trabajo. Solo puede especificar un flujo de trabajo de creación para la canalización. También puede omitir la creación por completo para configurar una canalización solo de prueba.

Flujo de trabajo de prueba

Los flujos de trabajo de prueba se ejecutan durante la etapa `Test` del marco del flujo de trabajo. Puede especificar hasta diez flujos de trabajo de prueba para una canalización. También puede omitir las pruebas por completo si solo quiere que se cree la canalización.

Definición de grupos de prueba para los flujos de trabajo de prueba

Los flujos de trabajo de prueba se definen dentro de grupos de prueba. Puede ejecutar hasta diez flujos de trabajo de prueba para una canalización. Debe decidir si desea ejecutar los flujos de trabajo de prueba en un orden específico o ejecutar tantos como sea posible a la vez. La forma en que se ejecuten dependerá de cómo defina los grupos de prueba. En los siguientes escenarios se muestran varias formas de definir los flujos de trabajo de prueba.

> **Note**
>
> Si usa la consola para crear flujos de trabajo, le recomendamos que dedique tiempo a planificar cómo desea ejecutar los flujos de trabajo de prueba antes de definir los grupos de prueba. En la consola, puede agregar o eliminar flujos de trabajo y grupos de prueba, pero no puede reordenarlos.

Escenario 1: ejecución de un flujo de trabajo de prueba cada vez

Para ejecutar todos los flujos de trabajo de prueba uno a uno, puede configurar hasta diez grupos de prueba, cada uno con un único flujo de trabajo de prueba. Los grupos de prueba se ejecutan de uno en uno, en el orden en que se agreguen a la canalización. Esta es una forma de garantizar que los flujos de trabajo de prueba se ejecuten uno a uno y en un orden específico.

Escenario 2: ejecución de varios flujos de trabajo de prueba a la vez

Si el orden no importa y desea ejecutar a la vez tantos flujos de trabajo de prueba como sea posible, puede configurar un único grupo de prueba e incluir en él el máximo número de flujos de trabajo de prueba. Generador de imágenes iniciará hasta cinco flujos de trabajo de prueba al mismo tiempo e iniciará otros flujos de trabajo de prueba a medida que se vayan completando. Si su objetivo es ejecutar los flujos de trabajo de prueba lo más rápido posible, esta es una forma de hacerlo.

Escenario 3: mezcla y combinación

Si tiene un escenario mixto, en el que algunos flujos de trabajo de prueba pueden ejecutarse al mismo tiempo y otros deben ejecutarse por separado, puede configurar los grupos de prueba para lograr este objetivo. El único límite a la hora de configurar los grupos de prueba es la cantidad máxima de flujos de trabajo de prueba que se pueden ejecutar en la canalización

Establecimiento de los parámetros del flujo de trabajo en una canalización de Generador de imágenes desde la consola

Los parámetros del flujo de trabajo funcionan de la misma manera para los flujos de trabajo de creación que para los flujos de trabajo de prueba. Cuando cree o actualice una canalización, seleccione los flujos de trabajo de creación y prueba que desee incluir. Si ha definido parámetros en el documento de flujos de trabajo para un flujo de trabajo seleccionado, Generador de imágenes los mostrará en el panel Parámetros. El panel está oculto para los flujos de trabajo que no tienen definidos parámetros.

En cada parámetro se muestran los siguientes atributos definidos en el documento de flujos de trabajo:

- Nombre (no editable): nombre del parámetro.
- Tipo (no editable): el tipo de datos del valor del parámetro.
- Valor: el valor del parámetro. Puede editar el valor del parámetro para configurarlo para la canalización.

Especificación del rol de servicio de IAM que Generador de imágenes utiliza para ejecutar las acciones de flujo de trabajo

Para ejecutar flujos de trabajo de imágenes, Generador de imágenes necesita permiso para realizar acciones de flujo de trabajo. Puede especificar el rol vinculado al servicio AWSServiceRoleForImageBuilder, o bien puede especificar su propio rol personalizado para el acceso al servicio, de la siguiente manera.

- Consola: en el paso 3 del asistente de canalización Definir el proceso de creación de imágenes, seleccione el rol vinculado al servicio o su propio rol personalizado de la lista Rol de IAM del panel Acceso al servicio.
- API Image Builder: en la solicitud de CreateImageacción, especifique el rol vinculado al servicio o su propio rol personalizado como valor del parámetro. `executionRole`

Para obtener más información sobre cómo crear un rol de servicio, consulte Crear un rol para delegar permisos a un AWS servicio en la Guía del AWS Identity and Access Management usuario.

Usa EventBridge reglas con los pipelines de Image Builder

Los eventos de una amplia gama de servicios AWS y de socios se transmiten a los autobuses de EventBridge eventos de Amazon casi en tiempo real. También puede generar eventos personalizados y enviarlos desde sus propias aplicaciones a EventBridge. Los buses de eventos usan reglas para determinar dónde enrutar los datos del evento.

Las canalizaciones de Image Builder están disponibles como objetivos de EventBridge reglas, lo que significa que puede ejecutar una canalización de Image Builder en función de las reglas que cree para responder a eventos en el bus o según una programación.

Para obtener un resumen de los eventos generados por el sistema a los que Image Builder envía EventBridge, consulteMensajes de eventos que envía Generador de imágenes.

> **Note**
>
> Los buses de eventos son específicos de una región. La regla y el objetivo deben estar en la misma región.

Contenido

- EventBridge términos
- Consulta EventBridge las reglas de tu pipeline de Image Builder
- Usa EventBridge reglas para programar la construcción de un oleoducto

EventBridge términos

Esta sección contiene un resumen de los términos para ayudarle a entender cómo EventBridge se integra en sus procesos de creación de Image Builder.

Evento

Describe un cambio en un entorno que puede afectar a uno o más recursos de la aplicación. El entorno puede ser un AWS entorno, un servicio o una aplicación de un socio de SaaS o una de

sus aplicaciones o servicios. También puede configurar eventos programados en una línea de tiempo.

Bus de eventos

Una canalización que recibe datos de eventos de aplicaciones y servicios.

Origen

El servicio o la aplicación que envió el evento al bus de eventos.

Destino

Un recurso o punto final que se EventBridge invoca cuando coincide con una regla y entrega los datos del evento al destino.

Regla

Una regla hace coincidir eventos de entrada y los dirige a destinos para procesamiento. Una sola regla puede enviar un evento a varios destinos, que luego pueden ejecutarse en paralelo. Las reglas se basan en un patrón de eventos o en un cronograma.

Patrón

Un patrón de eventos define la estructura del evento y los campos con los que coincide una regla para iniciar la acción objetivo.

Programación

Las reglas de programación realizan una acción según un cronograma, como ejecutar una canalización de Generador de Imágenes para actualizar una imagen trimestralmente. Existen dos tipos de expresiones de programación:

- Expresiones Cron: coinciden con criterios de programación específicos mediante la sintaxis cron, que puede describir criterios simples; por ejemplo, ejecutar semanalmente un día específico. También puede establecer criterios más complejos, como que se ejecute trimestralmente el quinto día del mes, entre las 2 de la mañana y las 4 de la mañana.
- Expresiones de frecuencia: especifican un intervalo regular cuando se invoque el objetivo, por ejemplo, cada 12 horas.

Consulta EventBridge las reglas de tu pipeline de Image Builder

La pestaña de EventBridge reglas de la página de detalles de las canalizaciones de Image Builder Image muestra los buses de EventBridge eventos a los que tiene acceso su cuenta y las reglas del

bus de eventos seleccionado que se aplican a la canalización actual. Esta pestaña también enlaza directamente con la EventBridge consola para crear nuevos recursos.

Acciones que enlazan con la EventBridge consola

- Creación de un bus de eventos
- Creación de reglas

Para obtener más información EventBridge, consulta los siguientes temas en la Guía del EventBridge usuario de Amazon.

- Qué es Amazon EventBridge
- Autobuses para EventBridge eventos de Amazon
- EventBridge Eventos de Amazon
- EventBridge Reglas de Amazon

Usa EventBridge reglas para programar la construcción de un oleoducto

Para este ejemplo, creamos una nueva regla de programación para el bus de eventos predeterminado mediante una expresión de frecuencia. La regla de este ejemplo genera un evento en el bus de eventos cada 90 días. El evento inicia la creación de una canalización para actualizar la imagen.

1. Abra la consola EC2 de Image Builder en https://console.aws.amazon.com/imagebuilder/.
2. Para ver una lista de las canalizaciones de imágenes creadas en su cuenta, elija canalizaciones de imágenes en el panel de navegación.

   > **Note**
   >
   > La lista de canalizaciones de imágenes incluye un indicador del tipo de imagen de salida que crea la canalización: AMI o Docker.

3. Para ver los detalles o editar una canalización, elija el enlace con el nombre de la canalización. Esto abre la vista detallada de la canalización.

> **Note**
>
> También puede seleccionar la casilla junto al nombre de la canalización, y, a continuación, Ver detalles.

4. Abra la pestaña de EventBridge reglas.
5. Mantenga el bus de eventos predeterminado que está preseleccionado en el panel del bus de eventos.
6. Elija Crear regla. Esto lo llevará a la página Crear regla en la EventBridge consola de Amazon.
7. Escriba un nombre y una descripción para la regla. El nombre de la regla debe ser único en el bus de eventos de la región seleccionada.
8. En el panel Definir patrón, elija la opción Programar. Esto expande el panel, con la Frecuencia fija para todas las opciones seleccionadas.
9. Introduzca `90` en el primer cuadro y seleccione Días en la lista desplegable.
10. Realice las siguientes acciones en el panel Seleccionar objetivos:
    a. Seleccione `EC2 Image Builder` en la lista desplegable Objetivo.
    b. Para aplicar la regla a una canalización de Generador de Imágenes, seleccione la canalización de destino en la lista desplegable de Canalización de imágenes.
    c. EventBridge necesita permiso para iniciar una compilación para la canalización seleccionada. Para este ejemplo, mantenga la opción predeterminada de Crear un nuevo rol para este recurso específico.
    d. Elija Agregar objetivo.
11. Elija Creación.

> **Note**
>
> Para obtener más información sobre la configuración de las reglas de expresión de tasas que no se tratan en este ejemplo, consulte Expresiones de tarifas en la Guía del EventBridge usuario de Amazon.

Integración de productos y servicios en Generador de imágenes

EC2 Image Builder se integra con AWS Marketplace otras aplicaciones para ayudarle a crear imágenes de máquinas personalizadas robustas Servicios de AWS y seguras.

Productos

Las recetas de Image Builder pueden incorporar productos de imagen AWS Marketplace y componentes gestionados por Image Builder para proporcionar funciones especializadas de creación y prueba, de la siguiente manera.

- AWS Marketplace productos de imagen: utilice un producto de imagen AWS Marketplace como imagen base en su receta para cumplir con los estándares de la organización, como el endurecimiento del CIS. Al crear una receta desde la consola de Generador de Imágenes, puede elegir entre las suscripciones existentes o buscar un producto específico de AWS Marketplace. Al crear una receta desde la API, la CLI o el SDK de Generador de Imágenes, puede especificar el nombre de recurso de Amazon (ARN) de un producto de imagen para usarlo como su imagen base.
- Componentes de Generador de imágenes: los componentes que especifique en sus recetas pueden realizar acciones de compilación y prueba, por ejemplo, para instalar software o realizar una validación de conformidad. Algunos productos de imagen a los que se suscribe desde AWS Marketplace pueden incluir un componente complementario que puede usar en sus recetas. Las imágenes de CIS Hardened incluyen un TOE de AWS componente correspondiente que puede utilizar en su receta para aplicar las directrices de nivel 1 de CIS Benchmarks para su configuración.

> **Note**
>
> Si desea más información sobre los productos relacionados con la conformidad, consulte Productos de conformidad para sus imágenes de Generador de Imágenes.

Servicios

Image Builder se integra con lo siguiente Servicios de AWS para proporcionar métricas, registros y monitoreo detallados de eventos. Esta información le ayuda a realizar un seguimiento de su actividad, solucionar problemas de compilación de imágenes y crear automatizaciones basadas en las notificaciones de eventos.

- AWS Organizations— AWS Organizations le permite aplicar políticas de control de servicios (SCP) a las cuentas de su organización. Puede crear, administrar, habilitar y deshabilitar políticas individuales. Al igual que todos los demás AWS artefactos y servicios, Image Builder respeta las políticas definidas en AWS Organizations. AWS proporciona una plantilla SCPs para escenarios comunes, como imponer restricciones a las cuentas de los miembros para lanzar instancias que solo estén aprobadas AMIs.
- AWS CloudTrail— Supervise los eventos de Image Builder que se envían a CloudTrail. Para obtener más información sobre CloudTrail la integración con Image Builder, consulteRegistra las llamadas a la API Image Builder mediante CloudTrail.

  Para obtener más información sobre CloudTrail cómo activarla y buscar los archivos de registro, consulte la Guía del AWS CloudTrail usuario.
- Amazon CloudWatch Logs: supervise, almacene y acceda a sus archivos de registro de Image Builder con CloudWatch. De forma opcional, puede guardar sus registros en un bucket de S3. Para obtener más información sobre CloudWatch la integración con Image Builder, consulteSupervise los registros de Image Builder con Amazon CloudWatch Logs.

  Para obtener más información sobre CloudWatch los registros, consulta ¿Qué es Amazon CloudWatch Logs? en la Guía del usuario CloudWatch de Amazon Logs.
- Amazon Elastic Container Registry (Amazon ECR): Amazon ECR es un servicio de registro de imágenes de contenedor de AWS administrado que es seguro, escalable y fiable. Las imágenes de contenedor que crea con Generador de Imágenes se almacenan en Amazon ECR en su región de origen (donde se ejecuta la compilación) y en cualquier región en la que distribuya la imagen de contenedor. Para obtener más información sobre Amazon ECR, consulte la Guía del usuario de Amazon Elastic Container Registry.
- Amazon EventBridge: Conéctese a una transmisión de datos de eventos en tiempo real de las actividades de Image Builder en su cuenta. Para obtener más información EventBridge, consulta ¿Qué es Amazon EventBridge? en la Guía del EventBridge usuario de Amazon.
- Amazon Inspector: descubra vulnerabilidades en su software y en la configuración de red mediante escaneos automáticos para que la instancia de EC2 prueba que lance Image Builder cree una nueva imagen. Generador de imágenes guarda los resultados del recurso de imagen de salida

para que pueda investigarlos y corregirlos una vez que se termine la instancia de prueba. Para obtener más información sobre escaneos y precios, consulte ¿Qué es Amazon Inspector? en la Guía del usuario de Amazon Inspector.

Amazon Inspector también puede escanear sus repositorios de ECR si configura el escaneo mejorado. Para obtener más información, consulte Escaneo de imágenes de contenedor de Amazon ECR en la Guía del usuario de Amazon Inspector.

> **Note**
>
> Amazon Inspector es una característica de pago.

- AWS License Manager: puede adjuntar una licencia autoadministrada de License Manager a una AMI de salida durante el proceso de distribución. La licencia que especifique para la región de destino ya debe existir en esa región. Para obtener más información sobre las licencias autoadministradas, consulte Self-managed licenses in License Manager.
- AWS Marketplace: consulte una lista de las suscripciones de sus productos AWS Marketplace actuales y busque productos de imagen directamente desde Generador de Imágenes. También puede usar un producto de imagen al que se haya suscrito como imagen base para una receta de Generador de Imágenes. Para obtener más información sobre la administración de AWS Marketplace las suscripciones, consulta Comprar productos en la Guía del AWS Marketplace comprador.
- AWS Resource Access Manager (AWS RAM) — Con AWS RAM, puedes compartir recursos con cualquiera de ellos Cuenta de AWS o a través de ellos AWS Organizations. Si tienes varios Cuentas de AWS, puedes crear recursos de forma centralizada y usarlos AWS RAM para compartirlos con otras cuentas. EC2 Image Builder permite compartir los siguientes recursos: componentes, imágenes y recetas de imágenes. Para obtener más información al respecto AWS RAM, consulte la Guía AWS Resource Access Manager del usuario. Para obtener información sobre cómo compartir recursos de Generador de Imágenes, consulte Comparta los recursos de Image Builder con AWS RAM.
- Amazon Simple Notification Service (Amazon SNS): si está configurado, publique mensajes detallados sobre el estado de su imagen en un tema de SNS al que esté suscrito. Para obtener más información sobre Amazon SNS, consulte ¿Qué es Amazon SNS? en la Guía para desarrolladores de Amazon Simple Notification Service.

Temas sobre integración de productos y servicios

- EventBridge Integración de Amazon en Image Builder
- Integración de Amazon Inspector en Generador de Imágenes
- AWS Marketplace integración en Image Builder
- Integración de Amazon SNS en Generador de Imágenes
- Productos de conformidad para sus imágenes de Generador de Imágenes

EventBridge Integración de Amazon en Image Builder

Amazon EventBridge es un servicio de bus de eventos sin servidor que puede utilizar para conectar su aplicación Image Builder con datos relacionados de otros Servicios de AWS. En EventBridge, una regla hace coincidir los eventos entrantes y los envía a los destinos para su procesamiento. Una sola regla puede enviar un evento a varios destinos, y luego estos eventos se ejecutan en paralelo.

Con él EventBridge, puede automatizar los eventos del sistema Servicios de AWS y responder automáticamente a ellos, como los problemas de disponibilidad de las aplicaciones o los cambios en los recursos. Los eventos de Servicios de AWS se envían casi EventBridge en tiempo real. Puede configurar reglas que reaccionen a los eventos entrantes para iniciar acciones. Por ejemplo, enviar un evento a una función Lambda cuando el estado de una EC2 instancia cambia de pendiente a en ejecución. Estos se denominan patrones. Para crear una regla basada en un patrón de eventos, consulta Cómo crear EventBridge reglas de Amazon que reaccionen a los eventos en la Guía del EventBridge usuario de Amazon.

Entre las acciones que se pueden iniciar automáticamente se incluyen las siguientes:

- Invoca una función AWS Lambda
- Invocar el comando Amazon EC2 Run
- Transmitir el evento a Amazon Kinesis Data Streams
- Active una máquina de AWS Step Functions estados
- Notificación sobre un tema de Amazon SNS o una cola de Amazon SQS

También puede configurar reglas de programación para que el bus de eventos predeterminado realice una acción a intervalos regulares, como ejecutar una canalización de Generador de Imágenes para actualizar una imagen trimestralmente. Existen dos tipos de expresiones de programación:

- Expresiones cron: el siguiente ejemplo de una expresión cron programa una tarea para que se ejecute todos los días a mediodía UTC+0:

```
cron(0 12 * * ? *)
```

Para obtener más información sobre el uso de expresiones cron con EventBridge, consulte Expresiones cron en la Guía EventBridge del usuario de Amazon.

- Expresiones de frecuencia: el siguiente ejemplo de expresión de frecuencia programa una tarea para que se ejecute cada 12 horas:

```
rate(12 hour)
```

Para obtener más información sobre el uso de expresiones de tarifas con EventBridge, consulte Expresiones de tarifas en la Guía del EventBridge usuario de Amazon.

Para obtener más información sobre cómo se integran EventBridge las reglas con las canalizaciones de imágenes de Image Builder, consulteUsa EventBridge reglas con los pipelines de Image Builder.

Mensajes de eventos que envía Generador de imágenes

Image Builder envía mensajes de eventos EventBridge cuando se producen cambios significativos en el estado de los recursos de Image Builder. Por ejemplo, cuando se produce un cambio de estado en una imagen. En los ejemplos siguientes se muestran los mensajes de eventos JSON típicos que Generador de imágenes podría enviar.

Temas

- EC2 Cambio de estado de imagen de Image Builder
- EC2 Image Builder CVE detectado
- EC2 Paso a la espera del flujo de trabajo de Image Builder
- EC2 Image Builder Image Pipeline se deshabilita automáticamente

EC2 Cambio de estado de imagen de Image Builder

Generador de imágenes envía este evento cuando el estado de un recurso de imagen cambia durante la creación de la imagen. Por ejemplo, cuando el estado de la imagen cambia de un estado a otro, de la siguiente manera:

- Desde `building` hasta `testing`
- Desde `testing` hasta `distribution`

- Desde `testing` hasta `failed`
- Desde `integrating` hasta `available`

```
{
    "version": "0",
    "id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "detail-type": "EC2 Image Builder Image State Change",
    "source": "aws.imagebuilder",
    "account": "111122223333",
    "time": "2024-01-18T17:50:56Z",
    "region": "us-west-2",
    "resources": ["arn:aws:imagebuilder:us-west-2:111122223333:image/
cmkencryptedworkflowtest-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222/1.0.0/1"],
    "detail": {
        "previous-state": {
            "status": "TESTING"
        },
        "state": {
            "status": "AVAILABLE"
        }
    }
}
```

EC2 Image Builder CVE detectado

Si tiene habilitada la detección de CVE para la imagen, Generador de imágenes envía un mensaje con los resultados cada vez que se completa el escaneo de una imagen.

```
{
    "version": "0",
    "id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "detail-type": "EC2 Image Builder CVE Detected",
    "source": "aws.imagebuilder",
    "account": "111122223333",
    "time": "2023-03-01T16:59:09Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:imagebuilder:us-east-1:111122223333:image/test-image/1.0.0/1",
        "arn:aws:imagebuilder:us-east-1:111122223333:image-pipeline/test-pipeline"
    ],
    "detail": {
```

```
        "resource-id": "i-1234567890abcdef0",
        "finding-severity-counts": {
            "all": 0,
            "critical": 0,
            "high": 0,
            "medium": 0
        }
    }
}
```

EC2 Paso a la espera del flujo de trabajo de Image Builder

Generador de imágenes envía un mensaje cuando un paso del flujo de trabajo `WaitForAction` se detiene para esperar a que se complete una acción asincrónica.

```
{
    "version": "0",
    "id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "detail-type": "EC2 Image Builder Workflow Step Waiting",
    "source": "aws.imagebuilder",
    "account": "111122223333",
    "time": "2024-01-18T16:54:44Z",
    "region": "us-west-2",
    "resources": ["arn:aws:imagebuilder:us-west-2:111122223333:image/
workflowstepwaitforactionwithvalidsnstopictest-a1b2c3d4-5678-90ab-cdef-
EXAMPLE22222/1.0.0/1", "arn:aws:imagebuilder:us-west-2:111122223333:workflow/build/
build-workflow-a1b2c3d4-5678-90ab-cdef-EXAMPLE33333/1.0.0/1"],
    "detail": {
        "workflow-execution-id": "wf-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
        "workflow-step-execution-id": "step-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "workflow-step-name": "TestAutoSNSStop"
    }
}
```

EC2 Image Builder Image Pipeline se deshabilita automáticamente

Si has configurado el `autoDisablePolicy` para tu canalización, Image Builder inhabilita la canalización y envía un mensaje de evento EventBridge cuando el número de errores consecutivos de ejecución de la canalización programados supere el número máximo permitido por la política.

```
{
```

```
    "version": "0",
    "id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "detail-type": "EC2 Image Builder Image Pipeline Automatically Disabled",
    "source": "aws.imagebuilder",
    "account": "111122223333",
    "time": "2025-09-18T16:54:44Z",
    "region": "us-west-2",
    "resources": ["arn:aws:imagebuilder:us-west-2:111122223333:image-pipeline/disabled-
image-pipeline-name"],
    "detail": {
        "consecutive-failures": "5"
    }
}
```

Integración de Amazon Inspector en Generador de Imágenes

Cuando activa el escaneo de seguridad con Amazon Inspector, escanea continuamente las imágenes de las máquinas y las instancias en ejecución en su cuenta para detectar vulnerabilidades en el sistema operativo y el lenguaje de programación. Si está activado, el análisis de seguridad es automático y Generador de imágenes puede guardar una instantánea de los resultados de las instancias de prueba al crear una imagen nueva. Amazon Inspector es un servicio de pago.

Cuando Amazon Inspector descubre vulnerabilidades en el software o en la configuración de la red, toma las siguientes medidas:

- Le notifica que se ha producido un resultado.
- Califica la gravedad del resultado. La clasificación de gravedad categoriza las vulnerabilidades para ayudarle a priorizar sus resultados e incluye los siguientes valores:
  - Sin clasificar
  - Informativo
  - Bajo
  - Medio
  - Alto
  - Crítica
- Proporciona información sobre el resultado y enlaces a recursos adicionales para obtener más detalles.
- Ofrece una guía de corrección para ayudarle a resolver los problemas que generaron el resultado.

Configuración de escaneos de seguridad

Si has activado Amazon Inspector para tu cuenta, Amazon Inspector escanea automáticamente las EC2 instancias que lanza Image Builder para crear y probar una nueva imagen. Esas instancias tienen una vida útil corta durante el proceso de compilación y prueba y, por lo general, sus resultados caducan en cuanto se cierran esas instancias. Para ayudarlo a investigar y corregir los resultados de la nueva imagen, tiene la opción de que Generador de imágenes guarde como instantánea los resultados que Amazon Inspector haya identificado para la instancia de prueba durante el proceso de creación.

Para configurar los escaneos de seguridad para su canalización, consulte Configure los escaneos de seguridad para las imágenes de Image Builder en Consola de administración de AWS.

Revise los resultados de seguridad.

En la consola de Generador de Imágenes, puede ver los resultados de seguridad de todos sus recursos de Generador de Imágenes en un solo lugar. Puede ver todos los resultados en la página Resultados de seguridad, en la sección Información general sobre seguridad, o puede agruparlos por vulnerabilidad, por canalización de imágenes o por imagen. De forma predeterminada, la consola muestra todos los resultados de seguridad. El panel de resumen de la opción Todos los resultados de seguridad muestra el número de resultados que tiene para cada nivel de gravedad. Para obtener más información, consulte Gestione los resultados de seguridad de las imágenes de Image Builder en Consola de administración de AWS.

Para obtener más información sobre los resultados de vulnerabilidades de Amazon Inspector, consulte Descripción de los resultados en Amazon Inspector en la Guía del usuario de Amazon Inspector.

AWS Marketplace integración en Image Builder

AWS Marketplace es un catálogo digital seleccionado en el que puede encontrar software, datos y servicios de terceros y suscribirse a ellos, que le ayudarán a crear soluciones que se adapten a las necesidades de su empresa. AWS Marketplace reúne a compradores autenticados y vendedores registrados con listados de software de categorías populares, como seguridad, redes, almacenamiento, aprendizaje automático y más.

Un AWS Marketplace vendedor puede ser un proveedor de software independiente (ISV), un distribuidor o una persona que tiene algo que ofrecer relacionado con AWS productos y servicios.

Cuando el vendedor envía un producto AWS Marketplace, define el precio del producto y los términos y condiciones de uso. Los compradores aceptan los precios, los términos y las condiciones de la oferta. Para obtener más información AWS Marketplace, consulta ¿Qué es? AWS Marketplace

AWS Marketplace funciones de integración

Image Builder se integra AWS Marketplace para proporcionar las siguientes funciones directamente desde la consola de Image Builder:

- Busque los productos de imagen que estén disponibles en AWS Marketplace.
- Busque productos de AWS Marketplace imagen que suministren componentes.
- Consulte una lista de sus suscripciones de AWS Marketplace productos actuales.
- Usa un producto de AWS Marketplace imagen al que te hayas suscrito como imagen base para una receta de Image Builder.
- Usa AWS Marketplace los componentes a los que te hayas suscrito en una receta de Image Builder.

Image Builder se integra AWS Marketplace para mostrar los productos y componentes de imagen a los que te has suscrito. También puede buscar productos y componentes de AWS Marketplace imagen en la página Descubra productos sin salir de la consola Image Builder.

La AMI de salida que crea Image Builder incluye los códigos de producto de los productos y componentes de AWS Marketplace imagen. Puede tener hasta cuatro códigos de producto para la imagen final personalizada.

AWS Marketplace suscripciones en Image Builder

La página Suscripciones de la AWS Marketplace sección de la consola de Image Builder le muestra una lista de los AWS Marketplace productos a los que está suscrito actualmente. Cada producto suscrito muestra los siguientes detalles:

- El nombre de producto. Está vinculada a la página de detalles del producto en AWS Marketplace. La página de detalles del producto al que se ha suscrito se abre en una nueva pestaña del navegador.
- El editor. Está vinculado a la página de detalles del editor en AWS Marketplace. Se abrirá la página de detalles del publicador en una pestaña nueva en su navegador.
- La versión a la que te has suscrito.

- Si el producto suscrito incluye algún componente asociado, Image Builder muestra un enlace a los detalles del componente.

En la parte superior de la página, puede buscar un producto específico por su nombre o puede hojear los resultados con los controles de paginación. Para usar un producto de imagen suscrito en una nueva receta, seleccione un producto suscrito y elija Crear nueva receta. Generador de Imágenes preselecciona el primer producto de la lista de forma predeterminada.

> **Note**
>
> Si busca un producto al que se acaba de suscribir y no lo ve en la lista, utilice el botón de actualización de la parte superior de la pestaña para actualizar los resultados. Puede que tarde unos minutos en aparecer una nueva suscripción en la lista.

Descubra los productos de AWS Marketplace imagen desde la consola Image Builder

Esta sección se centra en los productos con AWS Marketplace imágenes para utilizarlos como imagen base en su receta. En el caso de los productos que incluyen componentes de software asociados, puedes filtrar por el propietario del producto en la consola y en la API, el SDK y la CLI. Para obtener más información, consulte Enumeración de componentes de Generador de imágenes. Para obtener más información sobre cómo buscar AWS Marketplace componentes, suscribirse a ellos y usarlos, consulteUtilice AWS Marketplace componentes para personalizar la imagen.

Descubra productos

Para buscar un producto de AWS Marketplace imagen en la consola de Image Builder, sigue estos pasos:

1. Abra la consola EC2 de Image Builder en https://console.aws.amazon.com/imagebuilder/.
2. En el panel de navegación, selecciona Descubrir productos en la AWS Marketplacesección.
3. Puedes buscar productos con imágenes en la pestaña Productos con imágenes de la página Descubre productos.

   Image Builder filtra previamente los productos AWS Marketplace para centrarse en las imágenes de máquinas que puede utilizar en sus recetas de Image Builder. Para obtener más información

sobre AWS Marketplace la integración con Image Builder, elija la pestaña que coincida con lo que desea ver.

Esta pestaña contiene dos paneles. A la izquierda, el panel Refinar los resultados le ayuda a filtrar los resultados para encontrar los productos a los que desea suscribirse. A la derecha, el panel Buscar productos muestra los productos que cumplen sus criterios de filtrado y también le da la opción de buscar por nombre de producto.

Cómo refinar los resultados

La siguiente lista muestra solo algunos de los filtros que puede aplicar a su búsqueda de productos:

- Seleccione una o más categorías de productos, como software de infraestructura o machine learning.
- Elija los sistemas operativos para su producto de imagen o elija todos los productos para una plataforma de sistema operativo específica, por ejemplo, All Linux/Unix.
- Elija uno o más publicadores para mostrar sus productos disponibles. Seleccione el enlace Mostrar todo para ver todos los publicadores que tienen productos que se ajustan a los filtros que ha aplicado.

> **Note**
>
> Los nombres de los publicadores no están en orden alfabético. Por ejemplo, si busca un publicador específico, como `Center for Internet Security`, puede introducir parte del nombre en el cuadro de búsqueda situado en la parte superior del cuadro de diálogo Todos los publicadores. Debe escribir el nombre al completo, ya que una abreviatura, como `CIS`, es posible que no produzca los resultados que busca. También puede buscar los nombres de los publicadores página por página.

Las opciones de filtro son dinámicas. Cada elección que realice afectará a sus opciones para todas las demás categorías. Hay miles de productos disponibles AWS Marketplace, por lo que cuanto más pueda filtrar, más probabilidades tendrá de encontrar lo que busca.

Búsqueda de productos

Para buscar un producto específico por su nombre, puede introducir parte del nombre en la barra de búsqueda situada en la parte superior de este panel. El resultado de cada producto incluye los siguientes detalles:

- El nombre y el logotipo del producto. Ambos están enlazados a la página de detalles del producto en AWS Marketplace. Se abrirá la página de detalles en una pestaña nueva en su navegador. Desde allí, puede suscribirse al producto de imagen si quiere usarlo en una receta de Generador de Imágenes. Para obtener más información, consulte Compra de productos en la Guía del comprador de AWS Marketplace .

  Si te suscribes al producto de imagen en AWS Marketplace, vuelve a la pestaña Image Builder del navegador y actualiza la lista de productos de imagen suscritos para verla.

  > **Note**
  >
  > Puede que tarde unos minutos en aparecer la nueva suscripción.

- El nombre del publicador. Está enlazado a la página de detalles del editor en AWS Marketplace. Se abrirá la página de detalles del publicador en una pestaña nueva en su navegador.
- La versión de producto.
- La calificación por estrellas del producto y los enlaces directos a la sección de reseñas de la página de detalles del producto en AWS Marketplace. Se abrirá la página de detalles en una pestaña nueva en su navegador.
- Las primeras líneas de la descripción del producto.

Justo debajo de la barra de búsqueda, puede ver cuántos resultados ha generado su búsqueda y qué subconjunto de esos resultados se muestra actualmente. Puede usar los controles adicionales de la parte derecha del panel para ajustar la configuración del número de productos que se van a mostrar al mismo tiempo y el orden de clasificación que se debe aplicar a los resultados. También puede utilizar el control de paginación para hojear los resultados.

Usa un producto AWS Marketplace de imagen en las recetas de Image Builder

Abre la página Crear recetas y selecciona un producto de AWS Marketplace imagen para usarlo como imagen base, de la siguiente manera.

1. Abra la consola EC2 de Image Builder en https://console.aws.amazon.com/imagebuilder/.
2. En el panel de navegación, seleccione Recetas de imagen en la sección AWS Marketplace. Esto le muestra una lista de recetas de imagen que ha creado.
3. Seleccione Crear receta de imagen. Esto abre la página Crear receta.
4. Introduzca el nombre y la versión de su receta en la sección Detalles de la receta como de costumbre.
5. En la sección Imagen base, elija la opción de AWS Marketplace imágenes de . Aquí se muestra una lista de los productos de AWS Marketplace imagen a los que te has suscrito en la pestaña Suscripciones. Puede elegir su imagen de base de la lista.

   También puedes buscar otros productos de imagen que estén disponibles AWS Marketplace directamente en la AWS Marketplacepestaña. Seleccione Añadir productos o abra la pestaña AWS Marketplacedirectamente. Para obtener más información sobre cómo configurar los filtros y buscar en la AWS Marketplace, consulteDescubra los productos de AWS Marketplace imagen desde la consola Image Builder.
6. Introduzca los detalles restantes como de costumbre. Si alguna de las suscripciones de sus productos incluye componentes de compilación, puede seleccionarlos en la lista de componentes de compilación. Seleccione un elemento `AWS Marketplace` de la lista de tipos de propietarios de componentes `Third party managed` para verlos o seleccione el componente CIS.
7. Elija Crear receta.

   La imagen final puede contener hasta cuatro códigos de producto de productos y componentes de la AWS Marketplace imagen. Si la imagen base y los componentes seleccionados contienen más de cuatro códigos de producto, Image Builder devolverá un error al intentar crear la receta.

Integración de Amazon SNS en Generador de Imágenes

Amazon Simple Notification Service (Amazon SNS) es un servicio administrado con el que se ofrece la entrega de mensajes asíncronos de los publicadores a los suscriptores (también conocidos como productores y consumidores).

Puede especificar un tema de SNS en la configuración de la infraestructura. Al crear una imagen o ejecutar una canalización, Generador de Imágenes puede publicar mensajes detallados sobre el estado de su imagen en este tema. Cuando el estado de la imagen alcanza uno de los siguientes estados, Generador de Imágenes publica un mensaje:

- `AVAILABLE`
- `FAILED`

Para ver un ejemplo de mensaje de SNS de Generador de Imágenes, consulte Formato del mensaje de SNS. Si desea crear un nuevo tema de SNS, consulte Introducción a Amazon SNS en la Guía del desarrollador de Amazon Simple Notification Service.

Temas de SNS cifrados

Si su tema de SNS está cifrado, debe conceder permiso en la AWS KMS key política para que el rol de servicio Image Builder lleve a cabo las siguientes acciones:

- `kms:Decrypt`
- `kms:GenerateDataKey`

> **Note**
>
> Si su tema de SNS está cifrado, la clave que cifra este tema debe residir en la cuenta en la que se ejecuta el servicio de Generador de Imágenes. Generador de Imágenes no puede enviar notificaciones a temas de SNS que estén cifrados con claves de otras cuentas.

Ejemplo de adición de la política de claves de KMS

En el siguiente ejemplo se muestra la sección adicional que se agrega a la política de claves de KMS. Utilice el nombre de recurso de Amazon (ARN) para el rol vinculado al servicio de IAM que Generador de Imágenes creó en su cuenta cuando usted creó una imagen de Generador

de Imágenes por primera vez. Para obtener más información sobre el rol vinculado al servicio de Generador de Imágenes, consulte Uso de roles vinculados a servicios de IAM para Generador de imágenes.

```
{
  "Statement": [{
    "Effect": "Allow",
    "Principal": {
      "AWS": "arn:aws:iam::123456789012:role/aws-service-role/
imagebuilder.amazonaws.com/AWSServiceRoleForImageBuilder"
    },
    "Action": [
      "kms:GenerateDataKey*",
      "kms:Decrypt"
    ],
    "Resource": "*"
  }]
}
```

Puede usar uno de los métodos siguientes para obtener el ARN.

Consola de administración de AWS

Para obtener el ARN del rol vinculado al servicio que Image Builder creó en su cuenta desde Consola de administración de AWS, siga estos pasos:

1. Abra la consola de IAM en https://console.aws.amazon.com/iam/.
2. En el panel de navegación izquierdo, elija Roles.
3. Busque `ImageBuilder` y elija el siguiente nombre de rol en los resultados: `AWSServiceRoleForImageBuilder`. Aparecerá la página de detalles del rol.
4. Elija el icono junto al nombre de ARN para copiar el ARN en el portapapeles.

AWS CLI

Para obtener el ARN del rol vinculado al servicio que Image Builder creó en su cuenta desde AWS CLI, utilice el comando get-role de IAM, de la siguiente manera.

```
aws iam get-role --role-name AWSServiceRoleForImageBuilder
```

Muestra parcial de salida:

```
{
    "Role": {
        "Path": "/aws-service-role/imagebuilder.amazonaws.com/",
        "RoleName": "AWSServiceRoleForImageBuilder",
        ...
        "Arn": "arn:aws:iam::123456789012:role/aws-service-role/
imagebuilder.amazonaws.com/AWSServiceRoleForImageBuilder",
        ...
}
```

Formato del mensaje de SNS

Una vez que Generador de Imágenes publique un mensaje en su tema de Amazon SNS, otros servicios que se suscriban al tema pueden filtrar el formato del mensaje y determinar si cumple los criterios para tomar medidas adicionales. Por ejemplo, un mensaje de éxito podría iniciar una tarea para actualizar un almacén de parámetros de AWS Systems Manager o lanzar un flujo de trabajo de pruebas de conformidad externo para la AMI de salida.

En el siguiente ejemplo, se muestra la carga útil JSON de un mensaje típico que Generador de Imágenes publica cuando se completa la compilación de una canalización y crea una imagen de Linux.

```
{
  "versionlessArn": "arn:aws:imagebuilder:us-west-1:123456789012:image/example-linux-
image",
  "semver": 123794003928538027489912 4227,
  "arn": "arn:aws:imagebuilder:us-west-1:123456789012:image/example-linux-
image/1.0.0/3",
  "name": "example-linux-image",
  "version": "1.0.0",
  "type": "AMI",
  "buildVersion": 3,
  "state": {
    "status": "AVAILABLE"
  },
  "platform": "Linux",
  "imageRecipe": {
    "arn": "arn:aws:imagebuilder:us-west-1:123456789012:image-recipe/example-linux-
image/1.0.0",
    "name": "amjule-barebones-linux",
    "version": "1.0.0",
```

```
    "components": [
      {
        "componentArn": "arn:aws:imagebuilder:us-west-1:123456789012:component/update-
linux/1.0.2/1"
      }
    ],
    "platform": "Linux",
    "parentImage": "arn:aws:imagebuilder:us-west-1:987654321098:image/amazon-linux-2-
x86/2022.6.14/1",
    "blockDeviceMappings": [
      {
        "deviceName": "/dev/xvda",
        "ebs": {
          "encrypted": false,
          "deleteOnTermination": true,
          "volumeSize": 8,
          "volumeType": "gp2"
        }
      }
    ],
    "dateCreated": "Feb 24, 2021 12:31:54 AM",
    "tags": {
      "internalId": "1a234567-8901-2345-bcd6-ef7890123456",
      "resourceArn": "arn:aws:imagebuilder:us-west-1:123456789012:image-recipe/example-
linux-image/1.0.0"
    },
    "workingDirectory": "/tmp",
    "accountId": "462045008730"
  },
  "sourcePipelineArn": "arn:aws:imagebuilder:us-west-1:123456789012:image-pipeline/
example-linux-pipeline",
  "infrastructureConfiguration": {
    "arn": "arn:aws:imagebuilder:us-west-1:123456789012:infrastructure-configuration/
example-linux-infra-config-uswest1",
    "name": "example-linux-infra-config-uswest1",
    "instanceProfileName": "example-linux-ib-baseline-admin",
    "tags": {
      "internalId": "234abc56-d789-0123-a4e5-6b789d012c34",
      "resourceArn": "arn:aws:imagebuilder:us-west-1:123456789012:infrastructure-
configuration/example-linux-infra-config-uswest1"
    },
    "logging": {
      "s3Logs": {
        "s3BucketName": "amzn-s3-demo-bucket"
```

```
      }
    },
    "keyPair": "example-linux-key-pair-uswest1",
    "terminateInstanceOnFailure": true,
    "snsTopicArn": "arn:aws:sns:us-west-1:123456789012:example-linux-ibnotices-
uswest1",
    "dateCreated": "Feb 24, 2021 12:31:55 AM",
    "accountId": "123456789012"
  },
  "imageTestsConfigurationDocument": {
    "imageTestsEnabled": true,
    "timeoutMinutes": 720
  },
  "distributionConfiguration": {
    "arn": "arn:aws:imagebuilder:us-west-1:123456789012:distribution-configuration/
example-linux-distribution",
    "name": "example-linux-distribution",
    "dateCreated": "Feb 24, 2021 12:31:56 AM",
    "distributions": [
      {
        "region": "us-west-1",
        "amiDistributionConfiguration": {}
      }
    ],
    "tags": {
      "internalId": "345abc67-8910-12d3-4ef5-67a8b90c12de",
      "resourceArn": "arn:aws:imagebuilder:us-west-1:123456789012:distribution-
configuration/example-linux-distribution"
    },
    "accountId": "123456789012"
  },
  "dateCreated": "Jul 28, 2022 1:13:45 AM",
  "outputResources": {
    "amis": [
      {
        "region": "us-west-1",
        "image": "ami-01a23bc4def5a6789",
        "name": "example-linux-image 2022-07-28T01-14-17.416Z",
        "accountId": "123456789012"
      }
    ]
  },
  "buildExecutionId": "ab0cd12e-34fa-5678-b901-2c3456d789e0",
  "testExecutionId": "6a7b8901-cdef-234a-56b7-8cd89ef01234",
```

```
  "distributionJobId": "1f234567-8abc-9d0e-1234-fa56b7c890de",
  "integrationJobId": "432109b8-afe7-6dc5-4321-0ba98f7654e3",
  "accountId": "123456789012",
  "osVersion": "Amazon Linux 2",
  "enhancedImageMetadataEnabled": true,
  "buildType": "USER_INITIATED",
  "tags": {
    "internalId": "901e234f-a567-89bc-0123-d4e567f89a01",
    "resourceArn": "arn:aws:imagebuilder:us-west-1:123456789012:image/example-linux-
image/1.0.0/3"
  }
}
```

El siguiente ejemplo muestra la carga útil JSON para un mensaje típico que Generador de Imágenes publica para un fallo de compilación de canalización para una imagen de Linux.

```
{
  "versionlessArn": "arn:aws:imagebuilder:us-west-2:123456789012:image/my-example-
image",
  "semver": 123794003928538027489912423 1,
  "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image/my-example-image/1.0.0/7",
  "name": "My Example Image",
  "version": "1.0.0",
  "type": "AMI",
  "buildVersion": 7,
  "state": {
    "status": "FAILED",
    "reason": "Image Failure reason."
  },
  "platform": "Linux",
  "imageRecipe": {
    "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/my-example-
image/1.0.0",
    "name": "My Example Image",
    "version": "1.0.0",
    "description": "Testing Image recipe",
    "components": [
      {
        "componentArn": "arn:aws:imagebuilder:us-west-2:123456789012:component/my-
example-image-component/1.0.0/1"
      }
    ],
    "platform": "Linux",
```

```
    "parentImage": "ami-0cd12345db678d90f",
    "dateCreated": "Jun 21, 2022 11:36:14 PM",
    "tags": {
      "internalId": "1a234567-8901-2345-bcd6-ef7890123456",
      "resourceArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/my-
example-image/1.0.0"
    },
    "accountId": "123456789012"
  },
  "sourcePipelineArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/my-
example-image-pipeline",
  "infrastructureConfiguration": {
    "arn": "arn:aws:imagebuilder:us-west-2:123456789012:infrastructure-configuration/
my-example-infra-config",
    "name": "SNS topic Infra config",
    "description": "An example that will retain instances of failed builds",
    "instanceTypes": [
      "t2.micro"
    ],
    "instanceProfileName": "EC2InstanceProfileForImageBuilder",
    "tags": {
      "internalId": "234abc56-d789-0123-a4e5-6b789d012c34",
      "resourceArn": "arn:aws:imagebuilder:us-west-2:123456789012:infrastructure-
configuration/my-example-infra-config"
    },
    "terminateInstanceOnFailure": true,
    "snsTopicArn": "arn:aws:sns:us-west-2:123456789012:example-pipeline-notification-
topic",
    "dateCreated": "Jul 5, 2022 7:31:53 PM",
    "accountId": "123456789012"
  },
  "imageTestsConfigurationDocument": {
    "imageTestsEnabled": true,
    "timeoutMinutes": 720
  },
  "distributionConfiguration": {
    "arn": "arn:aws:imagebuilder:us-west-2:123456789012:distribution-configuration/my-
example-distribution-config",
    "name": "New distribution config",
    "dateCreated": "Dec 3, 2021 9:24:22 PM",
    "distributions": [
      {
        "region": "us-west-2",
        "amiDistributionConfiguration": {},
```

```
        "fastLaunchConfigurations": [
          {
            "enabled": true,
            "snapshotConfiguration": {
              "targetResourceCount": 2
            },
            "maxParallelLaunches": 2,
            "launchTemplate": {
              "launchTemplateId": "lt-01234567890"
            },
            "accountId": "123456789012"
          }
        ]
      }
    ],
    "tags": {
      "internalId": "1fecd23a-4f56-7f89-01e2-345678abbe90",
      "resourceArn": "arn:aws:imagebuilder:us-west-2:123456789012:distribution-
configuration/my-example-distribution-config"
    },
    "accountId": "123456789012"
  },
  "dateCreated": "Jul 5, 2022 7:40:15 PM",
  "outputResources": {
    "amis": []
  },
  "accountId": "123456789012",
  "enhancedImageMetadataEnabled": true,
  "buildType": "SCHEDULED",
  "tags": {
    "internalId": "456c78b9-0e12-3f45-afb6-7e89b0f1a23b",
    "resourceArn": "arn:aws:imagebuilder:us-west-2:123456789012:image/my-example-
image/1.0.0/7"
  }
}
```


Productos de conformidad para sus imágenes de Generador de Imágenes

Con los estándares de seguridad en constante evolución, puede ser un desafío mantener la conformidad y proteger a su organización de las ciberamenazas. Para garantizar que sus imágenes personalizadas cumplen con las normas y se mantienen así mediante actualizaciones automáticas

cuando los editores publican nuevas versiones, Image Builder se integra con los productos de AWS Marketplace conformidad y los componentes de Image Builder.

Generador de Imágenes se integra con los siguientes productos de conformidad:

- Refuerzo de las referencias del Center for Internet Security (CIS, Centro para la seguridad de Internet)

  Puede utilizar las imágenes reforzadas de CIS y los componentes de refuerzo de CIS relacionados para compilar imágenes personalizadas que cumplan las últimas directrices de nivel 1 de las referencias de CIS. Las imágenes endurecidas por CIS están disponibles en AWS Marketplace. Para obtener más información sobre cómo configurar y utilizar las imágenes reforzadas de CIS y los componentes de refuerzo, consulte la Guía de inicio rápido: Componentes de refuerzo de CIS para EC2 Image Builder en la wiki de seguridad de CIS.

  > **Note**
  >
  > Cuando se suscribe a una imagen reforzada de CIS, también obtiene acceso al componente de compilación asociado que ejecuta un script para aplicar las directrices de nivel 1 de las referencias de CIS a su configuración. Para obtener más información, consulte Componentes de endurecimiento de CIS.

- Guías de implementación técnica de seguridad (STIG)

  Para cumplir con las normas STIG, utilice componentes STIG gestionados por Amazon Ejecutor y orquestador de tareas de AWS (TOE de AWS) en sus recetas de Image Builder. Los componentes de STIG escanean su instancia de compilación para detectar errores de configuración y ejecutan un script de corrección para corregir los problemas que encuentren. No podemos garantizar la conformidad con STIG para las imágenes que compile con Generador de Imágenes. Debe trabajar con el equipo de conformidad de su organización para verificar que su imagen final cumpla con los requisitos. Para obtener una lista completa de los componentes de TOE de AWS STIG que puede usar en sus recetas de Image Builder, consulteComponentes de endurecimiento de STIG administrados por Amazon para Generador de imágenes.

Supervisión de eventos y registros en Generador de imágenes

Para mantener la fiabilidad, la disponibilidad y el rendimiento de las canalizaciones de EC2 Image Builder, es importante supervisar los eventos y los registros. Los eventos y los registros le ayudan a ver el panorama general y a profundizar en los detalles cuando se produce un error en una llamada a la API. Generador de Imágenes se integra con servicios que pueden enviar alertas e iniciar respuestas automatizadas cuando los eventos coinciden con los criterios que ha configurado.

En los temas siguientes se describen las técnicas de supervisión que puede utilizar a través de los servicios que se integran con Generador de Imágenes.

Supervisión de eventos y registros

- Registra las llamadas a la API Image Builder mediante CloudTrail
- Supervise los registros de Image Builder con Amazon CloudWatch Logs

Registra las llamadas a la API Image Builder mediante CloudTrail

EC2 Image Builder está integrado con AWS CloudTrail un servicio que proporciona un registro de las acciones realizadas por un usuario, un rol o un AWS servicio en Image Builder. CloudTrail captura todas las llamadas a la API de Image Builder como eventos. Las llamadas capturadas incluyen las llamadas desde la consola de Generador de Imágenes y las llamadas desde el código a las operaciones de la API de Generador de Imágenes. Si crea una ruta, puede habilitar la entrega continua de CloudTrail eventos a un bucket de Amazon S3, incluidos los eventos de Image Builder. Si no configura una ruta, podrá ver los eventos más recientes en la CloudTrail consola, en el historial de eventos. Con la información recopilada por CloudTrail, puede determinar la solicitud que se realizó a Image Builder, la dirección IP desde la que se realizó la solicitud, quién la hizo, cuándo se realizó y detalles adicionales.

Para obtener más información CloudTrail, consulte la Guía AWS CloudTrail del usuario.

Información sobre Image Builder en CloudTrail

CloudTrail está habilitada en tu cuenta Cuenta de AWS al crear la cuenta. Cuando se produce una actividad en Image Builder, esa actividad se registra en un CloudTrail evento junto con otros eventos

de AWS servicio en el historial de eventos. Puede ver, buscar y descargar los eventos recientes en su Cuenta de AWS. Para obtener más información, consulte Visualización de eventos con el historial de CloudTrail eventos.

Para tener un registro continuo de tus eventos Cuenta de AWS, incluidos los eventos de Image Builder, crea una ruta. Un rastro permite CloudTrail entregar 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. La ruta registra los eventos de todas las regiones de la AWS partición y envía los archivos de registro al bucket de Amazon S3 que especifique. Además, puede configurar otros AWS servicios para analizar más a fondo los datos de eventos recopilados en los CloudTrail registros y actuar en función de ellos. Para más información, consulte los siguientes temas:

- Introducción a la creación de registros de seguimiento
- CloudTrail servicios e integraciones compatibles
- Configuración de las notificaciones de Amazon SNS para CloudTrail
- Recibir archivos de CloudTrail registro de varias regiones y recibir archivos de CloudTrail registro de varias cuentas

Todas las acciones de Image Builder se registran CloudTrail y documentan en la Referencia de la API de EC2 Image Builder. Por ejemplo, las llamadas a las `StartImagePipelineExecution` acciones y `CreateImagePipelineUpdateInfrastructureConfiguration`, generan entradas en los archivos de CloudTrail registro.

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 credenciales de usuario root o 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 AWS servicio.

Para obtener más información, consulte el elemento userIdentity de CloudTrail .

Supervise los registros de Image Builder con Amazon CloudWatch Logs

CloudWatch La compatibilidad con los registros está activada de forma predeterminada. Image Builder conserva los registros de compilación de la instancia durante el proceso de compilación y, a continuación, los transmite a CloudWatch Logs cuando se completa la compilación. Antes de crear la imagen de compilación final, Image Builder elimina los registros locales de la instancia.

Registros de creación de imágenes

Los registros de creación de imágenes contienen los registros detallados de una creación de imagen individual. Image Builder escribe los registros de creación de imágenes en el siguiente grupo y flujo de CloudWatch registros de Image Builder:

LogGroup: `/aws/imagebuilder/`*`ImageName`*

LogStream (x.x.x/x): *`ImageVersion/ImageBuildVersion`*

Registros de ejecución de canalización

Image Builder escribe registros de ejecución de la canalización cuando se ejecuta una canalización de imágenes o cuando una canalización omite una ejecución programada habitual, junto con detalles sobre por qué ocurrió eso. Por ejemplo, cuando las actualizaciones de dependencias se configuran para una canalización programada, los siguientes escenarios tendrán una mensajería asociada:

- Image Builder escribe una entrada de registro cuando omite la ejecución de la canalización porque no hubo actualizaciones de dependencias.
- Cuando hay una actualización de dependencias, Image Builder registra el ARN del recurso que ha cambiado.

Image Builder escribe los registros de ejecución de la canalización en el siguiente grupo y flujo de CloudWatch registros de Image Builder:

LogGroup: `/aws/imagebuilder/pipeline/`*`pipeline-name`*

LogStream: *`2025/09/01`* (la fecha de ejecución de la canalización en YYYY/MM/DD formato)

Cada registro de canalización se adjunta a la transmisión de ese día.

Grupos de registros personalizados

Si especificó grupos de registros personalizados para el registro de imágenes o canalizaciones, Image Builder escribe los registros en el LogGroup que especificó en su canalización o en uno de los comandos create image si ejecutó la compilación manualmente. Para obtener más información sobre cómo funciona esto en Image Builder, consulteConfigure el registro de canalizaciones.

cancelación de la suscripción

Para excluirse de la transmisión de CloudWatch registros, elimine los siguientes permisos asociados a la función personalizada que corresponda a su caso de uso.

- registros: CreateLogGroup
- registros: CreateLogStream
- registros: PutLogEvents

Tipo de registro	Función personalizada
Creación de imágenes	Rol de ejecución
Ejecución de canalizaciones	Rol de ejecución
Registros de componentes	Rol de perfil de instancia

Para la solución avanzada de problemas, puede ejecutar comandos y scripts predefinidos mediante Ejecución de comandos de AWS Systems Manager. Para obtener más información, consulte Solucionar problemas de Generador de imágenes.

Seguridad en Generador de imágenes

La seguridad en la nube AWS es la máxima prioridad. Como AWS cliente, usted se beneficia de una arquitectura de centro de datos y red diseñada para cumplir con los requisitos de las organizaciones más sensibles a la seguridad.

La seguridad es una responsabilidad compartida entre usted AWS y usted. El modelo de responsabilidad compartida describe esto como seguridad de la nube y seguridad en la nube:

- Seguridad de la nube:AWS es responsable de proteger la infraestructura que se ejecuta Servicios de AWS en la AWS nube.AWS también le proporciona servicios que puede utilizar de forma segura. Los auditores externos prueban y verifican periódicamente la eficacia de nuestra seguridad como parte de los AWS programas de de . Para obtener más información sobre los programas de conformidad que se aplican a EC2 Image Builder, consulte Servicios de AWS Scope by Compliance Program AWS .
- Seguridad en la nube: su responsabilidad viene determinada por el AWS servicio que utilice. También es responsable de otros factores, incluida la confidencialidad de los datos, los requisitos de la empresa y la legislación y la normativa aplicables.

Esta documentación le ayuda a comprender cómo aplicar el modelo de responsabilidad compartida cuando se utiliza Generador de Imágenes. En los siguientes temas, se le mostrará cómo configurar Generador de Imágenes para satisfacer sus objetivos de seguridad y conformidad. También aprenderá a utilizar otros recursos Servicios de AWS que le ayuden a supervisar y proteger sus recursos de Image Builder.

Temas

- La protección de datos y el modelo de responsabilidad AWS compartida en Image Builder
- Integración de Identity and Access Management para Generador de imágenes
- Recursos de validación de conformidad para Generador de imágenes
- Redundancia y resiliencia de datos en Generador de imágenes
- Seguridad de la infraestructura en Generador de Imágenes
- Administración de parches para imágenes de Generador de imágenes
- Prácticas recomendadas de seguridad para Generador de imágenes

La protección de datos y el modelo de responsabilidad AWS compartida en Image Builder

El modelo de responsabilidad AWS compartida modelo se aplica a la protección de datos en EC2 Image Builder. Como se describe en este modelo,AWS es responsable de proteger la infraestructura global en la que se ejecutan todos los 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, consulte las Preguntas frecuentes sobre la privacidad de datos. Para obtener información sobre la protección de datos en Europa, consulte 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, le recomendamos que proteja Cuenta de AWS las credenciales y configure los usuarios 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.
- Se utiliza SSL/TLS para comunicarse con AWS los recursos. Exigimos TLS 1.2 y recomendamos TLS 1.3.
- Configure la API y el registro de actividad de los usuarios con AWS CloudTrail. Para obtener información sobre el uso de CloudTrail senderos para capturar AWS actividades, consulte Cómo trabajar con CloudTrail senderos en la Guía del AWS CloudTrail usuario.
- Utilice soluciones de AWS cifrado, junto con todos los controles de seguridad predeterminados 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 por FIPS 140-3 para acceder a AWS través de una interfaz de línea de comandos o una API, utilice un punto final FIPS. Para obtener más información sobre los puntos de conexión de FIPS disponibles, consulte 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. Esto incluye cuando trabaja con Image Builder u otro Servicios de AWS mediante la consola, la API o AWS SDKs.AWS CLI Cualquier dato que introduzca en etiquetas o campos de formato libre utilizados para los 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 y administración de claves en Generador de imágenes

Generador de imágenes cifra los datos en tránsito y en reposo de forma predeterminada con una clave de KMS propiedad del servicio, excepto en los siguientes casos:

- Componentes personalizados: Generador de imágenes cifra los componentes personalizados con la clave de KMS predeterminada o una clave de KMS propiedad del servicio.
- Flujos de trabajo de imágenes: Generador de imágenes puede cifrar los flujos de trabajo de imágenes con una clave administrada por el cliente si especifica la clave durante la creación del flujo de trabajo. Generador de imágenes gestiona el cifrado y descifrado con la clave para ejecutar los flujos de trabajo que configuró para las imágenes.

Puede administrar sus propias claves a través de AWS KMS. Sin embargo, no tiene permiso para administrar la clave KMS de Generador de Imágenes de propiedad de Generador de Imágenes. Para obtener más información sobre cómo administrar las claves de KMS con AWS Key Management Service ellas, consulte Primeros pasos en la Guía para AWS Key Management Service desarrolladores.

Contexto de cifrado

Para proporcionar una comprobación adicional de integridad y autenticidad de los datos cifrados, tiene la opción de incluir un contexto de cifrado al cifrar los datos. Cuando un recurso se cifra con un contexto de cifrado, vincula AWS KMS criptográficamente el contexto al texto cifrado. El recurso solo se puede descifrar si el solicitante proporciona una coincidencia exacta, con distinción de mayúsculas y minúsculas, para el contexto.

En los ejemplos de políticas de esta sección se utiliza un contexto de cifrado similar al nombre de recurso de Amazon (ARN) de un recurso de flujo de trabajo de Generador de imágenes.

Cifrado de flujos de trabajo de imágenes con una clave administrada por el cliente

Para agregar una capa de protección, puede cifrar los recursos de flujo de trabajo de Generador de imágenes con su propia clave administrada por el cliente. Si utiliza la clave administrada por el cliente para cifrar los flujos de trabajo de Generador de imágenes que cree, debe conceder el acceso en la política de claves para que Generador de imágenes utilice su clave al cifrar y descifrar los recursos de flujo de trabajo. Puede revocar el acceso en cualquier momento. Sin embargo, si revoca el acceso a la clave, Generador de imágenes no tendrá acceso a ningún flujo de trabajo que ya esté cifrado.

El proceso para conceder acceso a Generador de imágenes para usar la clave administrada por el cliente consta de los dos pasos siguientes:

Paso 1: incorporación de permisos de política de claves para los flujos de trabajo de Generador de imágenes

Para permitir que Generador de imágenes cifre y descifre los recursos de flujo de trabajo al crear o utilizar esos flujos de trabajo, debe especificar los permisos en la política de claves de KMS.

En este ejemplo de política de claves se concede acceso a las canalizaciones de Generador de imágenes para cifrar los recursos de flujo de trabajo durante el proceso de creación y para descifrar los recursos de flujo de trabajo para utilizarlos. La política también concede acceso a los administradores de claves. El contexto de cifrado y la especificación de los recursos utilizan un comodín para abarcar todas las regiones en las que se disponen de recursos de flujo de trabajo.

Como requisito previo para utilizar flujos de trabajo de imágenes, creó un rol de ejecución de flujos de trabajo de IAM que concede permiso a Generador de imágenes para ejecutar acciones de flujo de trabajo. La entidad principal de la primera instrucción que se muestra en este ejemplo de política de claves debe especificar el rol de ejecución de flujos de trabajo de IAM.

Para obtener más información sobre las claves administradas por el cliente, consulte Managing access to customer managed keys en la Guía para desarrolladores de AWS Key Management Service.

JSON

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

```
  {
   "Sid": "Allow access to build images with encrypted workflow",
   "Effect": "Allow",
   "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/YourImageBuilderExecutionRole"
   },
   "Action": [
    "kms:Decrypt",
    "kms:GenerateDataKey"
   ],
   "Resource": "*",
   "Condition": {
    "StringLike": {
     "kms:EncryptionContext:aws:imagebuilder:arn":
 "arn:aws:imagebuilder:*:111122223333:workflow/*"
    }
   }
  },
  {
   "Sid": "Allow access for key administrators",
   "Effect": "Allow",
   "Principal": {
    "AWS": "arn:aws:iam::111122223333:root"
   },
   "Action": [
    "kms:*"
   ],
   "Resource": "arn:aws:kms:*:111122223333:key/*"
  }
 ]
}
```

Paso 2: concesión de acceso a la clave al rol de ejecución de flujos de trabajo

El rol de IAM que Generador de imágenes asume para ejecutar los flujos de trabajo necesita permiso para usar la clave administrada por el cliente. Sin acceso a la clave, Generador de imágenes no podrá cifrar ni descifrar los recursos de flujos de trabajo.

Edite la política del rol de ejecución de flujos de trabajo para agregar la siguiente instrucción de política.

JSON

```
{
 "Version":"2012-10-17",
 "Statement": [
  {
   "Sid": "AllowAccessToWorkflowKey",
   "Effect": "Allow",
   "Action": [
    "kms:Decrypt",
    "kms:GenerateDataKey"
   ],
   "Resource": "arn:aws:kms:us-west-2:111122223333:key/key_ID",
   "Condition": {
    "StringLike": {
     "kms:EncryptionContext:aws:imagebuilder:arn":
 "arn:aws:imagebuilder:*:111122223333:workflow/*"
    }
   }
  }
 ]
}
```

AWS CloudTrail eventos para flujos de trabajo de imágenes

Los siguientes ejemplos muestran AWS CloudTrail las entradas típicas para cifrar y descifrar los flujos de trabajo de imágenes que se almacenan con una clave gestionada por el cliente.

Ejemplo: GenerateDataKey

En este ejemplo, se muestra el aspecto que puede tener un CloudTrail evento cuando Image Builder invoca la acción de AWS KMSGenerateDataKey API desde la acción de CreateWorkflow API de Image Builder. Generador de imágenes debe cifrar un nuevo flujo de trabajo antes de crear el recurso de flujos de trabajo.

```
{
 "eventVersion": "1.08",
 "userIdentity": {
  "type": "AssumedRole",
  "principalId": "PRINCIPALID1234567890:workflow-role-name",
  "arn": "arn:aws:sts::111122223333:assumed-role/Admin/workflow-role-name",
```

```
   "accountId": "111122223333",
   "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
   "sessionContext": {
    "sessionIssuer": {
     "type": "Role",
     "principalId": "PRINCIPALID1234567890",
     "arn": "arn:aws:iam::111122223333:role/Admin",
     "accountId": "111122223333",
     "userName": "Admin"
    },
    "webIdFederationData": {},
    "attributes": {
     "creationDate": "2023-11-21T20:29:31Z",
     "mfaAuthenticated": "false"
    }
   },
   "invokedBy": "imagebuilder.amazonaws.com"
  },
  "eventTime": "2023-11-21T20:31:03Z",
  "eventSource": "kms.amazonaws.com",
  "eventName": "GenerateDataKey",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "imagebuilder.amazonaws.com",
  "userAgent": "imagebuilder.amazonaws.com",
  "requestParameters": {
   "encryptionContext": {
    "aws:imagebuilder:arn": "arn:aws:imagebuilder:us-west-2:111122223333:workflow/build/
sample-encrypted-workflow/1.0.0/*",
    "aws-crypto-public-key": "key value"
   },
   "keyId": "arn:aws:kms:us-west-2:111122223333:alias/ExampleKMSKey",
   "numberOfBytes": 32
  },
  "responseElements": null,
  "requestID": "a1b2c3d4-5678-90ab-cdef-EXAMPLEaaaaa",
  "eventID": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "readOnly": true,
  "resources": [
   {
    "accountId": "111122223333",
    "type": "AWS::KMS::Key",
    "ARN": "arn:aws:kms:us-west-2:111122223333:key/a1b2c3d4-5678-90ab-cdef-EXAMPLEzzzzz"
   }
  ],
```

```
 "eventType": "AwsApiCall",
 "managementEvent": true,
 "recipientAccountId": "111122223333",
 "eventCategory": "Management"
}
```

Ejemplo: Decrypt

En este ejemplo, se muestra el aspecto que puede tener un CloudTrail evento cuando Image Builder invoca la acción de AWS KMSDecrypt API desde la acción de GetWorkflow API de Image Builder. Las canalizaciones de Generador de imágenes tienen que descifrar un recurso de flujo de trabajo antes de poder usarlo.

```
{
 "eventVersion": "1.08",
 "userIdentity": {
  "type": "AssumedRole",
  "principalId": "PRINCIPALID1234567890:workflow-role-name",
  "arn": "arn:aws:sts::111122223333:assumed-role/Admin/workflow-role-name",
  "accountId": "111122223333",
  "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
  "sessionContext": {
   "sessionIssuer": {
    "type": "Role",
    "principalId": "PRINCIPALID1234567890",
    "arn": "arn:aws:iam::111122223333:role/Admin",
    "accountId": "111122223333",
    "userName": "Admin"
   },
   "webIdFederationData": {},
   "attributes": {
    "creationDate": "2023-11-21T20:29:31Z",
    "mfaAuthenticated": "false"
   }
  },
  "invokedBy": "imagebuilder.amazonaws.com"
 },
 "eventTime": "2023-11-21T20:34:25Z",
 "eventSource": "kms.amazonaws.com",
 "eventName": "Decrypt",
 "awsRegion": "us-west-2",
 "sourceIPAddress": "imagebuilder.amazonaws.com",
 "userAgent": "imagebuilder.amazonaws.com",
```

```
 "requestParameters": {
  "keyId": "arn:aws:kms:us-west-2:111122223333:key/a1b2c3d4-5678-90ab-cdef-
EXAMPLEzzzzz",
  "encryptionAlgorithm": "SYMMETRIC_DEFAULT",
  "encryptionContext": {
   "aws:imagebuilder:arn": "arn:aws:imagebuilder:us-west-2:111122223333:workflow/build/
sample-encrypted-workflow/1.0.0/*",
   "aws-crypto-public-key": "ABC123def4567890abc12345678/90dE/F123abcDEF+4567890abc123D
+ef1=="
  }
 },
 "responseElements": null,
 "requestID": "a1b2c3d4-5678-90ab-cdef-EXAMPLEbbbbb",
 "eventID": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
 "readOnly": true,
 "resources": [
  {
   "accountId": "111122223333",
   "type": "AWS::KMS::Key",
   "ARN": "arn:aws:kms:us-west-2:111122223333:key/a1b2c3d4-5678-90ab-cdef-EXAMPLEzzzzz"
  }
 ],
 "eventType": "AwsApiCall",
 "managementEvent": true,
 "recipientAccountId": "111122223333",
 "eventCategory": "Management"
}
```


Almacenamiento de datos en Generador de imágenes

Generador de Imágenes no almacena ninguno de sus registros en el servicio. Todos los registros se guardan en la EC2 instancia de Amazon que se utiliza para crear la imagen o en los registros de automatización de Systems Manager.

Privacidad del tráfico entre redes en Generador de imágenes

Las conexiones están protegidas entre Image Builder y las ubicaciones locales, entre AZs dentro de una AWS región y entre AWS regiones a través de HTTPS. No hay conexiones directas entre las cuentas.

Integración de Identity and Access Management para Generador de imágenes

Temas

- Público
- Autenticación con identidades
- Funcionamiento del Generador de imágenes con las políticas y roles de IAM
- Gestione los perímetros de datos para el acceso a la descarga de cubos de S3 en Image Builder
- Políticas basadas en identidades de Generador de Imágenes
- Permisos de IAM para flujos de trabajo personalizados
- Políticas basadas en recursos de Generador de Imágenes
- Utilice políticas AWS administradas para EC2 Image Builder
- Uso de roles vinculados a servicios de IAM para Generador de imágenes
- Solución de problemas de IAM en Generador de imágenes

Público

La forma de usar AWS Identity and Access Management(IAM) varía según la función que desempeñe:

- Usuario del servicio: solicite permisos al administrador si no puede acceder a las características (consulte Solución de problemas de IAM en Generador de imágenes).
- Administrador del servicio: determine el acceso de los usuarios y envíe las solicitudes de permiso (consulte Funcionamiento del Generador de imágenes con las políticas y roles de IAM).
- Administrador de IAM: escribe las políticas para administrar el acceso (consulte Políticas basadas en identidades de Generador de Imágenes).

Autenticación con identidades

Para obtener información detallada sobre cómo autenticar a las personas y los procesos de su empresa Cuenta de AWS, consulte Identidades en la Guía del usuario de IAM.

Funcionamiento del Generador de imágenes con las políticas y roles de IAM

Antes de utilizar IAM para administrar el acceso a Generador de Imágenes, conozca qué características de IAM se pueden utilizar con Generador de Imágenes.

Para obtener una visión general de cómo funcionan Image Builder y otros AWS servicios con la mayoría de las funciones de IAM, consulte AWS los servicios que funcionan con IAM en la Guía del usuario de IAM.

Políticas basadas en identidad para Generador de Imágenes

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 la identidad, consulte Definición de permisos de IAM personalizados con políticas administradas por el cliente 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 la política de JSON de IAM en la Guía del usuario de IAM.

Ejemplos de políticas basadas en identidad para Generador de Imágenes

Para ver ejemplos de políticas basadas en identidad de Generador de Imágenes, consulte Políticas basadas en identidades de Generador de Imágenes.

Políticas basadas en recursos dentro de Generador de Imágenes

Compatibilidad con las políticas basadas en recursos: sí

Las políticas basadas en recursos son documentos de política JSON que se asocian a un recurso. Los ejemplos de políticas basadas en recursos son las políticas de confianza de roles de IAM y las políticas de bucket de Amazon S3. En los servicios que admiten políticas basadas en recursos, los

administradores de servicios pueden utilizarlos para controlar el acceso a un recurso específico. Para el recurso al que se asocia la política, la política define qué acciones puede realizar una entidad principal especificada en ese recurso y en qué condiciones. Debe especificar una entidad principal en una política basada en recursos. Los principales pueden incluir cuentas, usuarios, roles, usuarios federados o.Servicios de AWS

Para habilitar el acceso entre cuentas, puede especificar toda una cuenta o entidades de IAM de otra cuenta como la entidad principal de una política en función de recursos. Para obtener más información, consulte Acceso a recursos entre cuentas en IAM en la Guía del usuario de IAM.

Acciones de política para Generador de Imágenes

Compatibilidad con las acciones de políticas: sí

Los administradores pueden usar las políticas de AWS JSON para especificar quién tiene acceso a qué. Es decir, qué entidad principal puede realizar acciones en qué recursos y en qué condiciones.

El elemento `Action` de una política JSON describe las acciones que puede utilizar para conceder o denegar el acceso en una política. Incluya acciones en una política para conceder permisos y así llevar a cabo la operación asociada.

Para ver una lista de las acciones de Image Builder, consulte Acciones definidas por EC2 Image Builder en la Referencia de autorización de servicios.

Las acciones de política de Generador de Imágenes utilizan el siguiente prefijo antes de la acción:

```
imagebuilder
```

Para especificar varias acciones en una única instrucción, sepárelas con comas.

```
"Action": [
 "imagebuilder:action1",
 "imagebuilder:action2"
 ]
```

Para ver ejemplos de políticas basadas en identidad de Generador de Imágenes, consulte Políticas basadas en identidades de Generador de Imágenes.

Recursos de políticas para Generador de Imágenes

Compatibilidad con los recursos de políticas: sí

Los administradores pueden usar las políticas de AWS JSON 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). En el caso de las acciones que no admiten permisos por recurso, utilice un carácter comodín (*) para indicar que la instrucción se aplica a todos los recursos.

```
"Resource": "*"
```

Para ver una lista de los tipos de recursos de Image Builder y sus tipos ARNs, consulte Recursos definidos por EC2 Image Builder en la Referencia de autorización de servicio. Para saber con qué acciones puede especificar el ARN de cada recurso, consulte Acciones definidas por EC2 Image Builder.

Para ver ejemplos de políticas basadas en identidad de Generador de Imágenes, consulte Políticas basadas en identidades de Generador de Imágenes.

Claves de condición de política para Generador de Imágenes

Compatibilidad con claves de condición de políticas específicas del servicio: sí

Los administradores pueden usar las políticas de AWS JSON 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 AWS globales, consulte las claves de contexto de condición AWS globales en la Guía del usuario de IAM.

Para ver una lista de claves de condición de Image Builder, consulte Claves de condición de EC2 Image Builder en la Referencia de autorización de servicio. Para saber con qué acciones y recursos puede utilizar una clave de condición, consulte Acciones definidas por EC2 Image Builder.

Para ver ejemplos de políticas basadas en identidad de Generador de Imágenes, consulte Políticas basadas en identidades de Generador de Imágenes.

ACLs en Image Builder

Soportes ACLs: No

Las listas de control de acceso (ACLs) controlan qué directores (miembros de la cuenta, usuarios o roles) tienen permisos para acceder a un recurso. ACLs son similares a las políticas basadas en recursos, aunque no utilizan el formato de documento de políticas JSON.

ABAC con Generador de Imágenes

Compatibilidad con ABAC (etiquetas en las políticas): parcial

El control de acceso basado en atributos (ABAC) es una estrategia de autorización que define permisos en función de atributos denominados etiquetas. Puede adjuntar etiquetas a las entidades y AWS los recursos de IAM y, a continuación, diseñar políticas de ABAC para permitir las operaciones cuando la etiqueta del director 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.

Uso de credenciales temporales con Generador de Imágenes

Compatibilidad con credenciales temporales: sí

Las credenciales temporales proporcionan acceso a AWS los recursos a corto plazo y se crean automáticamente cuando se utiliza la federación o se cambia de rol.AWS recomienda generar credenciales temporales de forma dinámica en lugar de utilizar 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.

Permisos de entidades principales entre servicios para Generador de Imágenes

Admite sesiones de acceso directo (FAS): sí

Las sesiones de acceso directo (FAS) utilizan los permisos del principal que llama y los que solicitan Servicio de AWS para realizar solicitudes a los servicios descendentes.Servicio de AWS Para obtener información sobre las políticas a la hora de realizar solicitudes de FAS, consulte Sesiones de acceso directo.

Roles de servicio para Generador de Imágenes

Compatible con roles de servicio: sí

Un rol de servicio es un rol de IAM que asume un servicio para realizar acciones en su nombre. Un administrador de IAM puede crear, modificar y eliminar un rol de servicio desde IAM. Para obtener más información, consulte Crear un rol para delegar permisos a un Servicio de AWS en la Guía del usuario de IAM.

> **Warning**
>
> Cambiar los permisos de un rol de servicio podría interrumpir la funcionalidad de Generador de Imágenes. Edite los roles de servicio solo cuando Generador de Imágenes proporcione orientación para hacerlo.

Roles vinculados a servicios para Generador de Imágenes

Compatible con roles vinculados al servicio: sí

Un rol vinculado a un servicio es un tipo de rol de servicio que está vinculado a un.Servicio de AWS El servicio puede asumir el rol para realizar una acción en su nombre. Los roles vinculados al servicio aparecen en usted 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 ver detalles sobre el rol vinculado a servicios de Generador de Imágenes, consulte Uso de roles vinculados a servicios de IAM para Generador de imágenes.

Políticas basadas en identidades de Generador de Imágenes

Con las políticas basadas en identidades de IAM, puede especificar las acciones permitidas o denegadas, así como los recursos y también las condiciones en las que se permiten o deniegan las acciones. Generador de Imágenes admite acciones, recursos y claves de condición específicos. Para obtener información sobre todos los elementos que se utilizan en una política de JSON, consulte

Acciones, recursos y claves de condición de Amazon EC2 Image Builder en la Guía del usuario de IAM.

Acciones

Las acciones de política de Generador de Imágenes utilizan el siguiente prefijo antes de la acción: `imagebuilder:`. Las instrucciones de la política deben incluir un elemento `Action` o un elemento `NotAction`. Generador de Imágenes define su propio conjunto de acciones que describen las tareas que se pueden realizar con este servicio.

Para especificar varias acciones en una única instrucción, sepárelas con comas del siguiente modo:

```
"Action": [
 "imagebuilder:action1",
 "imagebuilder:action2"
]
```

Puede utilizar caracteres comodín para especificar varias acciones (*). Por ejemplo, para especificar todas las acciones que comiencen con la palabra `List`, incluya la siguiente acción:

```
"Action": "imagebuilder:List*"
```

Para ver una lista de acciones de Generador de Imágenes, consulte Acciones, recursos y claves de condición de Servicios de AWS en la Guía del usuario de IAM.

Administración del acceso con políticas

Para obtener información detallada sobre cómo gestionar el acceso AWS mediante la creación de políticas y su vinculación a las identidades o los AWS recursos de IAM, consulte Políticas y permisos en la Guía del usuario de IAM.

El rol de IAM que asocie a su perfil de instancia debe tener permisos para ejecutar los componentes de compilación y prueba incluidos en su imagen. Las siguientes políticas del rol de IAM se deben asociar al rol de IAM que está asociado al perfil de instancia:

- EC2InstanceProfileForImageBuilder
- EC2InstanceProfileForImageBuilderECRContainerBuilds
- AmazonSSMManagedInstanceCore

Recursos

Los administradores pueden usar las políticas de AWS JSON 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). En el caso de las acciones que no admiten permisos por recurso, utilice un carácter comodín (*) para indicar que la instrucción se aplica a todos los recursos.

```
"Resource": "*"
```

El ARN se compone de varios nodos que ayudan a identificar el recurso y garantizar que el nombre sea único. Los últimos nodos del nombre incluyen varias variaciones de formato para el tipo, el nombre y el identificador del recurso. Cuando Image Builder crea un recurso, utiliza el siguiente formato:

`arn:aws:imagebuilder:`*`region`*`:`*`owner`*`:`*`resource-type/resource-name/version/build-version`*

> **Note**
>
> La versión de compilación no siempre se incluye en el ARN del recurso. Sin embargo, algunas operaciones de la API, por ejemplo GetComponent, necesitan la versión de compilación para identificar de forma exclusiva un recurso que se va a recuperar.

Para los recursos que Image Builder utiliza en sus recetas, como la imagen base o los componentes, el nodo propietario puede ser uno de los siguientes:

- El número de cuenta del propietario del recurso
- Para los recursos gestionados por Amazon: `aws`
- Para AWS Marketplace recursos: `aws-marketplace`

El siguiente ejemplo muestra el ARN de un componente gestionado para instalar el CloudWatch agente de Amazon en Linux:

```
arn:aws:imagebuilder:us-east-1:aws:component/amazon-cloudwatch-agent-linux/1.0.1/1
```

En este ejemplo se muestra el ARN de un componente gestionado ficticio de:AWS Marketplace

```
arn:aws:imagebuilder:us-east-1:aws-marketplace:component/example-linux-software-component/1.0.1
```

Para obtener más información sobre cómo obtener una lista de componentes, incluido el uso de un filtro de propiedad, consulte. Enumeración de componentes de Generador de imágenes

Ejemplo ARNs

Los siguientes son algunos ejemplos de recursos ARNs que puede especificar en una política de IAM:

- ARN de instancia

```
"Resource": "arn:aws:imagebuilder:us-east-1:111122223333:instance/i-1234567890abcdef0"
```

- Ejemplo de comodín (*) para especificar todas las instancias de una cuenta determinada

```
"Resource": "arn:aws:imagebuilder:us-east-1:111122223333:instance/*"
```

- Ejemplo de comodín (*) para especificar todas las versiones de un flujo de trabajo de imágenes gestionadas

```
"Resource": "arn:aws:imagebuilder:us-east-1:aws:workflow/build/build-image/*"
```

- ARN de imagen gestionada

```
"Resource": "arn:aws:imagebuilder:us-east-1:aws:image/amazon-linux-2-arm64/2024.12.17/1"
```

- Ejemplo de comodín (*) para especificar todas las versiones de una imagen gestionada

```
"Resource": "arn:aws:imagebuilder:us-east-1:aws:image/amazon-linux-2-arm64/x.x.x"
```

Muchas acciones de la API EC2 Image Builder implican varios recursos. Para especificar varios recursos en una sola sentencia, sepárelos ARNs con comas.

```
"Resource": [
```

```
    "resource1",
    "resource2"
]
```

Claves de condición

Generador de Imágenes proporciona claves de condición específicas del servicio, y admite el uso de algunas claves de condición globales. Para ver todas las claves de condición AWS globales, consulte las claves de contexto de condición AWS globales en la Guía del usuario de IAM. Se proporcionan las siguientes claves de condición específicas del servicio.

generador de imágenes: CreatedResourceTagKeys

Funciona con operadores de cadena.

Utilice esta clave para filtrar el acceso en función de la presencia de claves de etiqueta en la solicitud. Esto le permite administrar los recursos que crea Generador de Imágenes.

Disponibilidad: esta clave solo está disponible para y. `CreateInfrastrucutreConfiguration` `UpdateInfrastructureConfiguration` APIs

generador de imágenes:/CreatedResourceTag<key>

Funciona con operadores de cadena.

Utilice esta clave para filtrar el acceso en función de los pares clave-valor de etiqueta adjuntados al recurso creado por Generador de Imágenes. Esto le permite administrar los recursos de Generador de Imágenes mediante etiquetas definidas.

Disponibilidad: esta clave solo está disponible para y. `CreateInfrastrucutreConfiguration` `UpdateInfrastructureConfiguration` APIs

generador de imágenes: LifecyclePolicyResourceType

Funciona con operadores de cadena.

Use esta clave para filtrar el acceso por el tipo de recurso del ciclo de vida especificado en la solicitud.

El valor de esta clave puede ser `AMI_IMAGE` o`CONTAINER_IMAGE`.

Disponibilidad: esta clave solo está disponible para los caracteres `CreateLifecyclePolicy` y `UpdateLifecyclePolicy`APIs.

Creador de imágenes: EC2 MetadataHttpTokens

Funciona con operadores de cadena.

Utilice esta clave para filtrar el acceso según el requisito del token HTTP de los metadatos de la EC2 instancia especificado en la solicitud.

El valor de esta clave puede ser `optional` o `required`.

Disponibilidad: esta clave solo está disponible para `CreateInfrastrucutreConfiguration` y `UpdateInfrastructureConfiguration` APIs.

generador de imágenes: StatusTopicArn

Funciona con operadores de cadena.

Utilice esta clave para filtrar el acceso por el ARN de tema SNS en la solicitud en la que se publicarán las notificaciones de estado del terminal.

Disponibilidad: esta clave solo está disponible para y. `CreateInfrastrucutreConfiguration` `UpdateInfrastructureConfiguration` APIs

Ejemplos

Para ver ejemplos de políticas basadas en identidad de Generador de Imágenes, consulte Políticas basadas en identidades de Generador de Imágenes.

Políticas basadas en recursos de Generador de Imágenes

Las políticas basadas en recursos especifican qué acciones puede realizar una entidad principal especificada en el recurso de Generador de imágenes y en qué condiciones. Generador de Imágenes admite políticas de permisos basadas en recursos para los componentes, imágenes y recetas de imágenes. Las políticas basadas en recursos le permiten otorgar permiso de uso a otras cuentas por recurso. También puede utilizar una política basada en recursos para permitir que un AWS servicio acceda a sus componentes, imágenes y recetas de imágenes.

Para obtener información sobre cómo asociar una política basada en recursos a un componente, imagen o receta de imágenes, consulte Comparta los recursos de Image Builder con AWS RAM.

> **Note**
>
> Al actualizar una política de recursos usando Generador de Imágenes, la actualización aparecerá en la consola de RAM.

Autorización basada en etiquetas de Generador de Imágenes

Puede adjuntar etiquetas a los recursos de Generador de Imágenes o transferirlas en una solicitud a Generador de Imágenes. 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 `imagebuilder:ResourceTag/key-name`, `aws:RequestTag/key-name` o `aws:TagKeys`. Para obtener más información acerca del etiquetado de recursos de Generador de Imágenes, consulte Etiquete un recurso del AWS CLI.

Roles de IAM de Generador de Imágenes

Un rol de IAM es una entidad dentro de usted Cuenta de AWS que tiene permisos específicos.

Uso de credenciales temporales con Generador de Imágenes

Puede utilizar credenciales temporales para iniciar sesión con federación, asumir un rol de IAM o asumir un rol de acceso entre cuentas. Para obtener credenciales de seguridad temporales, puede llamar a operaciones de AWS STS API como AssumeRoleo GetFederationToken.

Roles vinculados a servicios

Los roles vinculados a un servicio permiten acceder Servicios de AWS a los recursos de otros servicios para completar una acción en tu nombre. Los roles vinculados a servicios aparecen en la cuenta de IAM y son propiedad del servicio. Un usuario con acceso administrativo puede ver, pero no editar, los permisos de los roles vinculados a servicios.

Generador de Imágenes admite roles vinculados a servicios. Para obtener información sobre cómo crear o administrar roles vinculados a servicios de Generador de Imágenes, consulte Uso de roles vinculados a servicios de IAM para Generador de imágenes.

Roles de servicio

Esta característica permite que un servicio asuma un rol de servicio en su nombre. Este rol permite que el servicio obtenga acceso a los recursos de otros servicios para completar una acción en su

nombre. Los roles de servicio aparecen en su cuenta de IAM y son propiedad de la cuenta. Esto significa que un usuario con acceso administrativo puede cambiar los permisos de este rol. Sin embargo, hacerlo podría deteriorar la funcionalidad del servicio.

Gestione los perímetros de datos para el acceso a la descarga de cubos de S3 en Image Builder

EC2 Image Builder mantiene dos clases de buckets S3 AWS propiedad del servicio que contienen los recursos descargables necesarios para ejecutar las cargas de trabajo de Image Builder en su cuenta. Si utiliza perímetros de datos para controlar el acceso a Amazon S3 en su entorno, es posible que necesite permitir explícitamente el acceso a estos depósitos. Puede usar el ARN del bucket o la URL del bucket para incluir estos buckets en la lista de permitidos, en función de cómo controle el acceso a Amazon S3.

Scripts de arranque para la administración de componentes (obligatorio)

Este depósito de S3 contiene scripts de arranque para configurar la TOE de AWS aplicación en las EC2 instancias que se utilizan para crear imágenes. Image Builder requiere acceso para descargar los scripts que permiten crear y probar nuevas imágenes.

- ARN de la cubeta S3: `arn:<AWS partition>:s3:::ec2imagebuilder-managed-resources-<AWS Region>-prod`
- URL del depósito de S3: `https://ec2imagebuilder-managed-resources-<AWS Region>.s3.<AWS Region>.<AWS partition-specific domain name>`

Componentes gestionados

Este depósito de S3 contiene cargas útiles de paquetes para los componentes gestionados por Amazon. Image Builder requiere acceso para descargar todos los componentes gestionados que estén configurados en sus recetas.

- ARN de la cubeta S3: `arn:<AWS partition>:s3:::ec2imagebuilder-toe-<AWS Region>-prod`
- URL del depósito de S3: `https://ec2imagebuilder-toe-<AWS Region>.s3.<AWS Region>.<AWS partition-specific domain name>`

Políticas basadas en identidades de Generador de Imágenes

Temas

- Prácticas recomendadas de políticas basadas en identidad
- Uso de la consola de Generador de Imágenes

Prácticas recomendadas de políticas basadas en identidad

Las políticas basadas en identidad determinan si alguien puede crear, acceder o eliminar los recursos de Generador de Imágenes 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:

- Comience con las políticas AWS administradas y avance hacia los permisos con privilegios mínimos: para empezar a conceder permisos a sus usuarios y cargas de trabajo, utilice las políticas AWS administradas que otorgan permisos para muchos casos de uso comunes. Están disponibles en su.Cuenta de AWS Le recomendamos que reduzca aún más los permisos definiendo políticas administradas por el AWS cliente que sean específicas para sus casos de uso. 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, puede escribir una condición de políticas para especificar que todas las solicitudes deben enviarse utilizando SSL. También puedes usar condiciones para conceder el acceso a las acciones del servicio si se utilizan a través de una acción específica Servicio de AWS, 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.

- Requerir autenticación multifactor (MFA): si tiene un escenario que requiere usuarios de IAM o un usuario raíz en Cuenta de AWS su cuenta, active la MFA para 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 la consola de Generador de Imágenes

Para acceder a la consola de EC2 Image Builder, debe tener un conjunto mínimo de permisos. Estos permisos deben permitirle enumerar y ver los detalles de los recursos de Generador de Imágenes en su Cuenta de AWS. Si crea una política basada en identidad que sea más restrictiva que el mínimo de permisos necesarios, la consola no funcionará del modo esperado para las entidades (usuarios o roles de IAM) que tengan esa política.

Para garantizar que sus entidades de IAM puedan utilizar la consola de Image Builder, debe adjuntarles una de las siguientes políticas AWS gestionadas:

- Política AWSImageBuilderReadOnlyAccess
- Política AWSImageBuilderFullAccess

Para obtener más información sobre las políticas administradas de Generador de Imágenes, consulte Utilice políticas AWS administradas para EC2 Image Builder.

> **Important**
>
> La política AWSImageBuilderFullAccess es necesaria para crear el rol vinculado a servicios de Generador de Imágenes. Al asociar esta política a una entidad de IAM, también debe asociar la siguiente política personalizada e incluir los recursos que desee utilizar y que no tengan `imagebuilder` en el nombre del recurso:

JSON

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

```
        {
            "Effect": "Allow",
            "Action": [
                "sns:Publish"
            ],
            "Resource": "arn:aws:sns:*:*:*imagebuilder*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:GetInstanceProfile"
            ],
            "Resource": "arn:aws:iam::*:instance-profile/*imagebuilder*"
        },
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": [
                "arn:aws:iam::*:instance-profile/*imagebuilder*",
                "arn:aws:iam::*:role/*imagebuilder*"
            ],
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": "ec2.amazonaws.com"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:ListBucket"
            ],
            "Resource": "arn:aws:s3::*:*imagebuilder*"
        }
    ]
}
```

No es necesario que concedas permisos mínimos de consola a los usuarios que solo realicen llamadas a la API AWS CLI o a la AWS API. En su lugar, permite acceso únicamente a las acciones que coincidan con la operación de API que intenta realizar.

Permisos de IAM para flujos de trabajo personalizados

Cuando se utilizan flujos de trabajo personalizados con acciones escalonadas específicasRegisterImage, por ejemplo, es posible que se requieran permisos de IAM adicionales además de las políticas gestionadas por Image Builder estándar. En esta sección se describen los permisos adicionales necesarios para las acciones escalonadas personalizadas del flujo de trabajo.

RegisterImage permisos de acción escalonada

La acción del `RegisterImage` paso requiere EC2 permisos específicos de Amazon para registrar AMIs y, opcionalmente, recuperar etiquetas de instantáneas. Al utilizar el `includeSnapshotTags` parámetro, se necesitan permisos adicionales para describir las instantáneas.

Permisos necesarios para la acción RegisterImage escalonada:

Para todos los recursos, permita las siguientes acciones:

- `ec2:RegisterImage`
- `ec2:DescribeSnapshots`

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:RegisterImage",
                "ec2:DescribeSnapshots"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateTags"
            ],
            "Resource": "arn:aws:ec2:*::image/*",
            "Condition": {
                "StringEquals": {
                    "ec2:CreateAction": "RegisterImage"
```

```
                }
            }
        }
    ]
}
```

Detalles del permiso:

- `ec2:RegisterImage`- Necesario para registrar imágenes nuevas a AMIs partir de instantáneas
- `ec2:DescribeSnapshots`- Necesario cuando se utiliza `includeSnapshotTags: true` para recuperar etiquetas de instantáneas para fusionarlas con etiquetas AMI
- `ec2:CreateTags`- Necesario para aplicar etiquetas a la AMI registrada, incluidas las etiquetas predeterminadas de Image Builder y las etiquetas de instantáneas combinadas

> **Note**
>
> El `ec2:DescribeSnapshots` permiso solo se usa cuando el `includeSnapshotTags` parámetro está establecido en`true`. Si no usa esta función, puede omitir este permiso.

Comportamiento de combinación de etiquetas:

Cuando `includeSnapshotTags` está habilitada, la acción del RegisterImage paso hará lo siguiente:

- Recupere las etiquetas de la primera instantánea especificada en el mapeo de dispositivos de bloques
- Excluya todas las etiquetas AWS reservadas (aquellas con claves que comiencen por «aws:»)
- Combinar etiquetas de instantáneas con las etiquetas de registro AMI predeterminadas de Image Builder
- Dé prioridad a las etiquetas de Image Builder cuando las claves de etiquetas entren en conflicto

Políticas basadas en recursos de Generador de Imágenes

Para obtener información sobre cómo crear un componente, consulte Uso de componentes para personalizar su imagen de Generador de imágenes.

Restricción del acceso de los componentes de Generador de Imágenes a direcciones IP específicas

En el siguiente ejemplo se conceden permisos a cualquier usuario para que realice operaciones de Generador de Imágenes en los componentes. Sin embargo, la solicitud debe proceder del rango de direcciones IP especificado en la condición.

La condición de esta declaración identifica el rango 54.240.143.* de direcciones IP del Protocolo de Internet de la versión 4 (IPv4) permitidas, con una excepción: 54.240.143.188.

El `Condition` bloque utiliza las `NotIpAddress` condiciones `IpAddress` y y la clave de condición, que es una clave de condición que abarca todo el espacio`aws:SourceIp`.AWS Para obtener más información sobre estas claves de condición, consulte Especificación de condiciones en una política. Los `aws:sourceIp` IPv4 valores utilizan la notación CIDR estándar. Para obtener más información, consulte Operadores de condición de dirección IP en la guía del usuario de IAM.

JSON

```
{
  "Version":"2012-10-17",
  "Id": "IBPolicyId1",
  "Statement": [
    {
      "Sid": "IPAllow",
      "Effect": "Allow",
      "Action": "imagebuilder:GetComponent",
      "Resource": "arn:aws:imagebuilder:*::examplecomponent/*",
      "Condition": {
         "IpAddress": {"aws:SourceIp": "54.240.143.0/24"},
         "NotIpAddress": {"aws:SourceIp": "54.240.143.188/32"}
      }
    }
  ]
}
```

Utilice políticas AWS administradas para EC2 Image Builder

Una política AWS administrada es una política independiente creada y administrada por AWS.AWS Las políticas administradas están diseñadas para proporcionar permisos para muchos casos de uso comunes, de modo que pueda empezar a asignar permisos a usuarios, grupos y funciones.

Ten en cuenta que es posible que las políticas AWS administradas no otorguen permisos con privilegios mínimos para tus casos de uso específicos, ya que están disponibles para que los usen todos los AWS clientes. Se recomienda definir políticas administradas por el cliente específicas para sus casos de uso a fin de reducir aún más los permisos.

No puedes cambiar los permisos definidos en AWS las políticas administradas. Si AWS actualiza los permisos definidos en una política AWS administrada, la actualización afecta a todas las identidades principales (usuarios, grupos y roles) a las que está asociada la política.AWS es más probable que actualice una política AWS administrada cuando Servicio de AWS se lance una nueva o cuando estén disponibles nuevas operaciones de API para los servicios existentes.

Para obtener más información, consulte Políticas administradas por AWS en la Guía del usuario de IAM.

Política AWSImageBuilderFullAccess

La política AWSImageBuilderFullAccess otorga acceso total a los recursos de Generador de Imágenes de EC2 para el rol al que está asociado, lo que permite al rol enumerar, describir, crear, actualizar y eliminar los recursos de Generador de Imágenes. La política también otorga permisos específicos a los relacionados Servicios de AWS que sean necesarios, por ejemplo, para verificar los recursos o para mostrar los recursos actuales de la cuenta en Consola de administración de AWS.

Detalles de los permisos

Esta política incluye los permisos siguientes:

- Generador de Imágenes: se concede acceso administrativo para que el rol pueda enumerar, describir, crear, actualizar y eliminar los recursos de Generador de Imágenes de EC2.
- Amazon EC2: se concede acceso a las acciones de Amazon EC2 Describe necesarias para verificar la existencia de los recursos u obtener listas de los recursos que pertenecen a la cuenta.
- IAM: se concede acceso para obtener y utilizar perfiles de instancia cuyo nombre contenga “imagebuilder”, para comprobar la existencia del rol vinculado al servicio de Generador de Imágenes de EC2 mediante la acción de la API `iam:GetRole` y para crear el rol vinculado a servicios de Generador de Imágenes de EC2.

- License Manager: se concede acceso para enumerar las configuraciones de licencia o las licencias de un recurso.
- Amazon S3: se concede acceso a los buckets de listas que pertenecen a la cuenta y también a los buckets de Generador de Imágenes de EC2 con “imagebuilder” en sus nombres.
- Amazon SNS: se conceden permisos de escritura a Amazon SNS para verificar la propiedad de los temas que contienen “imagebuilder”.

Para ver los permisos de esta política, consulte AWSImageBuilderFullAccess en la Referencia de la política administrada de AWS.

Política AWSImageBuilderReadOnlyAccess

La política AWSImageBuilderReadOnlyAccess proporciona acceso de solo lectura a todos los recursos de Generador de Imágenes. Se conceden permisos para comprobar que el rol vinculado a servicios de Generador de Imágenes de EC2 existe mediante la acción de la API `iam:GetRole`.

Detalles de los permisos

Esta política incluye los permisos siguientes:

- Generador de Imágenes: se concede acceso para el acceso de solo lectura a los recursos de Generador de Imágenes.
- IAM: se concede acceso para verificar la existencia del rol vinculado a servicios de Generador de Imágenes de EC2 mediante la acción de la API `iam:GetRole`.

Para ver los permisos de esta política, consulte AWSImageBuilderReadOnlyAccess en la Referencia de la política administrada de AWS.

Política AWSServiceRoleForImageBuilder

La AWSServiceRoleForImageBuilderpolítica permite a Image Builder llamar Servicios de AWS en su nombre.

Detalles de los permisos

Esta política se asocia al rol vinculado a servicios de Generador de Imágenes de EC2 cuando el rol se crea a través de Systems Manager. Para obtener más información sobre el rol vinculado a servicios de Generador de Imágenes, consulte Uso de roles vinculados a servicios de IAM para Generador de imágenes.

La política incluye los siguientes permisos:

- CloudWatch Registros: se permite el acceso para crear y cargar CloudWatch registros en cualquier grupo de registros cuyo nombre comience por`/aws/imagebuilder/`.
- Amazon EC2: Image Builder tiene acceso para crear, tomar instantáneas y registrar las imágenes que cree (AMIs) y lanzar EC2 instancias en su cuenta. Image Builder utiliza instantáneas, volúmenes, interfaces de red, subredes, grupos de seguridad, configuración de licencias y pares de claves relacionados según sea necesario, siempre que la imagen, la instancia y los volúmenes que se estén creando o utilizando estén etiquetados con `CreatedBy: EC2 Image Builder` o. `CreatedBy: EC2 Fast Launch`

  Image Builder puede obtener información sobre EC2 las imágenes de Amazon, los atributos de las instancias, el estado de las instancias, los tipos de instancias disponibles para su cuenta, las plantillas de lanzamiento, las subredes, los hosts y las etiquetas de sus EC2 recursos de Amazon.

  Generador de Imágenes puede actualizar la configuración de imágenes para permitir o deshabilitar el inicio más rápido de las instancias de Windows en su cuenta, cuando la imagen tenga la etiqueta `CreatedBy: EC2 Image Builder`.

  Además, Generador de Imágenes puede iniciar, detener y finalizar las instancias que se ejecuten en su cuenta, compartir instantáneas de Amazon EBS, crear y actualizar imágenes y plantillas de lanzamiento, anular el registro de imágenes existentes, añadir etiquetas y replicar imágenes en cuentas a las que haya concedido permisos mediante la política Ec2ImageBuilderCrossAccountDistributionAccess. El etiquetado de Generador de Imágenes es necesario para todas estas acciones, como se describió anteriormente.
- Amazon ECR: se concede acceso a Generador de Imágenes para crear un repositorio, si es necesario, para escanear las vulnerabilidades de las imágenes de contenedor y etiquetar los recursos que crea para limitar el alcance de sus operaciones. También se concede acceso a Generador de Imágenes para eliminar las imágenes de contenedor que creó para los escaneos después de tomar instantáneas de las vulnerabilidades.
- EventBridge— Se concede acceso a Image Builder para crear y gestionar EventBridge reglas.
- IAM: se concede acceso a Image Builder para transferir cualquier rol de su cuenta a Amazon EC2 y a VM Import/Export.
- Amazon Inspector: se concede acceso a Generador de Imágenes para determinar cuándo Amazon Inspector completa los escaneos de las instancias de compilación y para recopilar los resultados de las imágenes que están configuradas para permitirlo.

- AWS KMS: se concede acceso a Amazon EBS para cifrar, descifrar o volver a cifrar los volúmenes de Amazon EBS. Esto es crucial para garantizar que los volúmenes cifrados funcionen cuando Generador de Imágenes compile una imagen.
- License Manager: se permite el acceso a Generador de Imágenes para actualizar las especificaciones de License Manager a través de `license-manager:UpdateLicenseSpecificationsForResource`.
- Amazon SNS: se conceden permisos de escritura para cualquier tema de Amazon SNS de la cuenta.
- Systems Manager: se concede acceso a Generador de imágenes para enumerar los comandos de Systems Manager, sus invocaciones y las entradas de inventario, describir la información de las instancias y los estados de ejecución de la automatización, describir los hosts para la compatibilidad con la ubicación de instancias y obtener los detalles de las invocaciones. Generador de Imágenes también puede enviar señales de automatización y detener las ejecuciones de automatización de cualquier recurso en su cuenta.

  Generador de Imágenes puede emitir invocaciones de comandos de ejecución en cualquier instancia que esté etiquetada con `"CreatedBy": "EC2 Image Builder"` para los siguientes archivos de script: `AWS-RunPowerShellScript`, `AWS-RunShellScript` o `AWSEC2-RunSysprep`. Generador de Imágenes puede iniciar una ejecución de automatización de Systems Manager en su cuenta para los documentos de automatización cuyo nombre comience con `ImageBuilder`.

  Generador de Imágenes también puede crear o eliminar asociaciones de State Manager para cualquier instancia en su cuenta, siempre que el documento de asociación sea `AWS-GatherSoftwareInventory`, y crear el rol vinculado a servicios de Systems Manager en su cuenta.

  Image Builder puede leer los parámetros públicos del almacén de parámetros y leer y actualizar los parámetros privados con el prefijo `/imagebuilder/` para poder actualizar el valor del parámetro con la AMI de salida IDs que Image Builder crea a partir de una nueva compilación.
- AWS STS: se concede acceso para que Generador de Imágenes asuma los roles denominados EC2ImageBuilderDistributionCrossAccountRole en su cuenta a cualquier cuenta en la que la política de confianza del rol lo permita. Esto se utiliza para la distribución de imágenes entre cuentas.

Para ver los permisos de esta política, consulte AWSServiceRoleForImageBuilder en la Referencia de la política administrada de AWS.

Política Ec2ImageBuilderCrossAccountDistributionAccess

La política Ec2ImageBuilderCrossAccountDistributionAccess concede permisos a Generador de Imágenes para distribuir imágenes entre las cuentas de las regiones de destino. Además, Image Builder puede describir, copiar y aplicar etiquetas a cualquier EC2 imagen de Amazon de la cuenta. La política también permite modificar los permisos de la AMI mediante la acción de la API `ec2:ModifyImageAttribute`.

Detalles de los permisos

Esta política incluye los permisos siguientes:

- Amazon EC2: Amazon tiene acceso EC2 para describir, copiar y modificar los atributos de una imagen y para crear etiquetas para cualquier EC2 imagen de Amazon de la cuenta.

Para ver los permisos de esta política, consulte Ec2ImageBuilderCrossAccountDistributionAccess en la Referencia de la política administrada de AWS.

Política EC2ImageBuilderLifecycleExecutionPolicy

La EC2ImageBuilderLifecycleExecutionPolicypolítica otorga permisos para que Image Builder lleve a cabo acciones como desaprobar, deshabilitar o eliminar los recursos de imagen de Image Builder y sus recursos subyacentes (AMIsinstantáneas) a fin de respaldar las reglas automatizadas para las tareas de administración del ciclo de vida de las imágenes.

Detalles de los permisos

Esta política incluye los permisos siguientes:

- Amazon EC2: se concede acceso EC2 a Amazon para realizar las siguientes acciones con Amazon Machine Images (AMIs) en la cuenta con la que están etiquetadas`CreatedBy: EC2 Image Builder`.
  - Habilitar o deshabilitar una AMI.
  - Habilitar o deshabilitar la obsolescencia de imágenes.
  - Describir una AMI y anular su registro.
  - Describir y modificar los atributos de imagen de la AMI.

  - Eliminar las instantáneas de volumen asociadas a la AMI.
  - Recuperar las etiquetas de un recurso.
  - Agregar o eliminar etiquetas de una AMI para su obsolescencia.
- Amazon ECR: se concede acceso a Amazon ECR para realizar las siguientes acciones por lotes en los repositorios de ECR con la etiqueta `LifecycleExecutionAccess: EC2 Image Builder`. Las acciones por lotes admiten reglas automatizadas del ciclo de vida de las imágenes de contenedor.
  - `ecr:BatchGetImage`
  - `ecr:BatchDeleteImage`

  El acceso se concede en el nivel de repositorio para los repositorios de ECR etiquetados con `LifecycleExecutionAccess: EC2 Image Builder`.
- AWS Grupos de recursos: se concede acceso a Image Builder para obtener recursos basados en etiquetas.
- EC2 Image Builder: se concede acceso a Image Builder para eliminar los recursos de imágenes de Image Builder.

Para ver los permisos de esta política, consulte EC2ImageBuilderLifecycleExecutionPolicy en la Referencia de la política administrada de AWS.

Política EC2InstanceProfileForImageBuilder

La EC2InstanceProfileForImageBuilderpolítica concede los permisos mínimos necesarios para que una EC2 instancia funcione con Image Builder. Esto no incluye los permisos necesarios para usar Systems Manager Agent.

Detalles de los permisos

Esta política incluye los permisos siguientes:

- CloudWatch Registros: se permite el acceso para crear y cargar CloudWatch registros en cualquier grupo de registros cuyo nombre comience por`/aws/imagebuilder/`.
- Amazon EC2: se concede acceso para describir volúmenes e instantáneas, para crear instantáneas de recursos de volúmenes o instantáneas que creó Image Builder y para crear etiquetas para los recursos de Image Builder.
- Image Builder: se permite el acceso a cualquier Image Builder o AWS Marketplace componente.

- AWS KMS— Se concede acceso para descifrar un componente de Image Builder, si se cifró mediante AWS KMS.
- Amazon S3: el acceso se concede para obtener objetos almacenados en un bucket de Amazon S3 cuyo nombre comience por`ec2imagebuilder-`, o recursos que tengan una extensión de archivo ISO.

Para ver los permisos de esta política, consulte EC2InstanceProfileForImageBuilder en la Referencia de la política administrada de AWS.

Política EC2InstanceProfileForImageBuilderECRContainerBuilds

La EC2InstanceProfileForImageBuilderECRContainerBuildspolítica concede los permisos mínimos necesarios para que una EC2 instancia cuando se trabaja con Image Builder cree imágenes de Docker y, a continuación, registre y almacene las imágenes en un repositorio de contenedores de Amazon ECR. Esto no incluye los permisos necesarios para usar Systems Manager Agent.

Detalles de los permisos

Esta política incluye los permisos siguientes:

- CloudWatch Registros: se permite el acceso para crear y cargar CloudWatch registros en cualquier grupo de registros cuyo nombre comience por. `/aws/imagebuilder/`
- Amazon ECR: se concede el acceso a Amazon ECR para obtener, registrar y almacenar una imagen de contenedor y para obtener un token de autorización.
- Generador de Imágenes: se permite el acceso para obtener un componente o receta de contenedor de Generador de Imágenes.
- AWS KMS— Se concede acceso para descifrar un componente de Image Builder o una receta de contenedor, si se cifró mediante AWS KMS.
- Amazon S3: se concede acceso para obtener objetos almacenados en un bucket de Amazon S3 cuyo nombre comience con `ec2imagebuilder-`.

Para ver los permisos de esta política, consulte EC2InstanceProfileForImageBuilderECRContainerBuilds en la Referencia de la política administrada de AWS.

Image Builder actualiza las políticas AWS gestionadas

En esta sección se proporciona información sobre las actualizaciones de las políticas AWS gestionadas de Image Builder desde que este servicio comenzó a realizar el seguimiento de estos cambios. Para recibir alertas automáticas sobre los cambios en esta página, suscríbase a la fuente RSS en la página del historial de documentos de Generador de Imágenes.

Cambio	Descripción	Fecha
AWSServiceRoleForImageBuilder: actualización de una política actual	Image Builder realizó los siguientes cambios en la función de servicio para permitir el uso de parámetros del almacén de parámetros AWS Systems Manager(SSM) en las recetas y durante la distribución de imágenes. • Se agregó ssm: GetParameter para permitir que Image Builder lea parámetros SSM públicos y parámetros SSM privados con el prefijo SSM `/imagebuilder/` para poder usarlos en recetas. • Se agregó ssm: PutParameter para permitir que Image Builder actualice los parámetros SSM privados `/imagebuilder/` con el prefijo de la AMI de salida que Image IDs Builder crea a partir de una nueva compilación.	23 de julio de 2025

Cambio	Descripción	Fecha
EC2InstanceProfileForImageBuilder: actualización de una política actual	Image Builder realizó los siguientes cambios en la política del perfil de la instancia para admitir más extensiones de archivo para las descargas de archivos ISO. • Se agregaron dos extensiones de archivo a la lista de recursos de `s3:GetObject` acción: `"arn:aws:s3:::*/*.iso"` y`"arn:aws:s3:::*/*.Iso"` .	19 de mayo de 2025
AWSServiceRoleForImageBuilder: actualización de una política actual	Image Builder realizó los siguientes cambios en la función de servicio para permitir la importación de archivos ISO del sistema operativo cliente de Microsoft como imagen base. • Se agregó ec2: RegisterImage para permitir a Image Builder crear una instantánea y crear y registrar una AMI cuyo sistema operativo base se importó de archivos de disco ISO verificados.	30 de diciembre de 2024

Cambio	Descripción	Fecha
EC2InstanceProfileForImageBuilder: actualización de una política actual	Image Builder realizó los siguientes cambios en la política del perfil de la instancia para permitir la creación de imágenes a partir de archivos de imagen de disco. • Se agregaron las siguientes acciones de ec2: DescribeVolumes y DescribeSnapshots en todos los recursos para recuperar detalles. También se agregaron las siguientes acciones para los recursos creados por Image Builder: CreateSnapshot para los recursos de volúmenes e instantáneas, y CreateTags. Se agregó s3: GetObject para recursos con una extensión de archivo «ISO».	30 de diciembre de 2024
EC2InstanceProfileForImageBuilder: política actualizada	Image Builder actualizó la `EC2InstanceProfileForImageBuilder` política para permitir a Image Builder obtener AWS Marketplace componentes.	2 de diciembre de 2024

Cambio	Descripción	Fecha
EC2ImageBuilderLifecycleExecutionPolicy: política nueva	Generador de imágenes agregó la nueva política `EC2ImageBuilderLifecycleExecutionPolicy` que contiene permisos para la administración del ciclo de vida de las imágenes.	17 de noviembre de 2023
AWSServiceRoleForImageBuilder: actualización de una política actual	Generador de imágenes hizo los siguientes cambios en el rol de servicio para brindar compatibilidad con la ubicación de instancias. • Se agregó ec2: DescribeHosts permite a Image Builder sondear el HostiD para determinar cuándo está en un estado válido para lanzar una instancia. • Se agregó la acción de API ssm:GetCommandInvocation, para mejorar el método que utiliza Image Builder para obtener detalles de la invocación de comandos.	19 de octubre de 2023

Cambio	Descripción	Fecha
AWSServiceRoleForImageBuilder: actualización de una política actual	Generador de imágenes hizo los siguientes cambios en el rol de servicio para brindar compatibilidad con la ubicación de instancias. • Se agregó ec2: DescribeHosts permite que Image Builder sondee el HostiD para determinar cuándo está en un estado válido para lanzar una instancia. • Se agregó la acción de API ssm:GetCommandInvocation, para mejorar el método que utiliza Image Builder para obtener detalles de la invocación de comandos.	28 de septiembre de 2023

Cambio	Descripción	Fecha
AWSServiceRoleForImageBuilder: actualización de una política actual	Generador de Imágenes realizó los siguientes cambios en el rol de servicio para permitir que los flujos de trabajo de Generador de Imágenes recopilen los resultados de vulnerabilidades para las compilaciones de imágenes de contenedor de AMI y ECR. Los nuevos permisos admiten la caracterí stica de detección e informes de CVE. • Se agregaron inspector2: ListCoverage e inspector2: para permitir a ListFindings Image Builder determinar cuándo Amazon Inspector completa los escaneos de las instancias de prueba y recopilar los resultados de las imágenes que están configuradas para permitirlo. • Se agregó ecr:Creat eRepository, con el requisito de que Image Builder etiquete el repositorio con `CreatedBy: EC2 Image Builder` (tag-on-create). También se agregó ecr: TagResource (obligato rio para tag-on-create)	30 de marzo de 2023

Cambio	Descripción	Fecha
	con la misma restricción de CreatedBy etiqueta, y una restricción adicional que requiere empezar por el nombre del repositor io. `image-builder-*` La restricción de nombre impide el escalado de privilegios y evita cambios en los repositorios que Generador de Imágenes no creó. • Se agregó ecr: BatchDele teImage para los repositor ios de ECR etiquetados con. `CreatedBy: EC2 Image Builder` Este permiso requiere que el nombre del repositorio empiece con `image-bui lder-*` . • Se han añadido permisos de eventos para que Image Builder cree y gestione las reglas EventBridge gestionadas por Amazon que incluyen `ImageBuil der-*` el nombre.	

Cambio	Descripción	Fecha
AWSServiceRoleForImageBuilder: actualización de una política actual	Generador de Imágenes hizo los siguientes cambios en el rol de servicio: • Se agregaron las licencias de License Manager como recurso para la RunInstance llamada ec2: para permitir a los clientes utilizar la imagen AMIs base asociada a una configuración de licencia.	22 de marzo de 2022
AWSServiceRoleForImageBuilder: actualización de una política actual	Generador de Imágenes hizo los siguientes cambios en el rol de servicio: • Se agregaron permisos para la acción de la EC2 EnableFastLaunch API, a fin de habilitar y deshabilitar el lanzamiento más rápido de las instancias de Windows. • Se ha reducido aún más el alcance de ec2: condiciones de etiquetas de CreateTags acción y recurso.	21 de febrero de 2022

Cambio	Descripción	Fecha
AWSServiceRoleForImageBuilder: actualización de una política actual	Generador de Imágenes hizo los siguientes cambios en el rol de servicio: • Se agregaron permisos para llamar al servicio VMIE para importar una VM y crear una AMI base a partir de ella. • Se ha reducido el alcance de ec2: condiciones de etiquetas de CreateTags acciones y recursos.	20 de noviembre de 2021
AWSServiceRoleForImageBuilder: actualización de una política actual	Generador de Imágenes agregó nuevos permisos para solucionar problemas donde más de una asociación de inventario provoca que la compilación de imágenes se bloquee.	11 de agosto de 2021

Cambio	Descripción	Fecha
AWSImageBuilderFullAccess: actualización de una política actual	Generador de Imágenes hizo los siguientes cambios en el rol de acceso completo: • Permisos añadidos para permitir `ec2:DescribeInstanceTypeOfferings` . • Se agregaron permisos para llamar a `ec2:DescribeInstanceTypeOfferings` para permitir que la consola de Generador de Imágenes refleje con precisión los tipos de instancias disponibles en la cuenta.	13 de abril de 2021
Generador de Imágenes comenzó el seguimiento de los cambios	Image Builder comenzó a realizar un seguimiento de los cambios de sus políticas AWS gestionadas.	2 de abril de 2021

Uso de roles vinculados a servicios de IAM para Generador de imágenes

EC2 Image Builder utiliza funciones AWS Identity and Access Management vinculadas a servicios (IAM). Un rol vinculado a servicios es un tipo único de rol de IAM que está vinculado directamente a Generador de Imágenes. Image Builder predefine las funciones vinculadas al servicio e incluyen todos los permisos que el servicio requiere para llamar a otros Servicios de AWS en su nombre.

Un rol vinculado a servicios hace que la configuración de Generador de Imágenes sea más eficiente, porque no tiene que añadir manualmente los permisos necesarios. Generador de Imágenes define los permisos de sus roles vinculados a servicios y, a menos que se defina de otra manera, solo

Generador de Imágenes puede asumir sus roles. Los permisos definidos incluyen la política de confianza y la política de permisos. La política de permisos no se puede asociar a ninguna otra entidad de IAM.

Para obtener información sobre otros servicios que admiten roles vinculados a servicios, consulte los Servicios de AWS que funcionan con IAM y busque los que muestran Sí en la columna Rol vinculado a servicios. Elija una opción Sí con un enlace para ver la documentación acerca del rol vinculado al servicio en cuestión.

Permisos de roles vinculados a servicios de Generador de Imágenes

Image Builder utiliza la función AWSServiceRoleForImageBuildervinculada al servicio para permitir que EC2 Image Builder acceda a AWS los recursos en su nombre. El rol vinculado a servicios confía en el servicio imagebuilder.amazonaws.com para asumir el rol.

No es necesario crear manualmente este rol vinculado a servicios. Cuando crea su primera imagen de Image Builder en la consola AWS de administración AWS CLI, la o la AWS API, Image Builder crea el rol vinculado al servicio por usted.

Las siguientes acciones crean una nueva imagen:

- Ejecutar el asistente de canalización en la consola de Generador de Imágenes para crear una imagen personalizada.
- Utilice una de las siguientes acciones de la API o el comando correspondiente AWS CLI:
  - La acción de la CreateImage API (create-imageen AWS CLI).
  - La acción de la ImportVmImage API (import-vm-imageen AWS CLI).
  - La acción de la StartImagePipelineExecution API (start-image-pipeline-executionen AWS CLI).

> **Important**
>
> Si se elimina de su cuenta el rol vinculado a servicios, puede utilizar el mismo proceso para volver a crearlo. Al crear su primer recurso de EC2 Image Builder, Image Builder vuelve a crear el rol vinculado al servicio para usted.

Para ver los permisos para AWSServiceRoleForImageBuilder, consulte la página Política AWSServiceRoleForImageBuilder. Para obtener más información sobre cómo configurar los

permisos para un rol vinculado a servicios, consulte Permisos de rol vinculado a servicios en la Guía del usuario de IAM.

Cómo eliminar un rol vinculado a servicios de Generador de Imágenes de su cuenta

Puede utilizar la consola de IAM AWS CLI, la o la AWS API para eliminar manualmente de su cuenta el rol vinculado al servicio de Image Builder. Sin embargo, antes de hacerlo, debe asegurarse de que no haya recursos de Generador de Imágenes habilitados que hagan referencia a este.

> **Note**
>
> Si el servicio de Generador de Imágenes está utilizando el rol cuando intenta eliminar los recursos, es posible que no se pueda eliminar. En tal caso, espere unos minutos e intente de nuevo la operación.

Limipieza de los recursos de Generador de Imágenes utilizados por el rol `AWSServiceRoleForImageBuilder`

1. Compruebe que no se esté ejecutando ninguna compilación de canalización antes de empezar. Para cancelar una compilación en ejecución, use el comando `cancel-image-creation` de la AWS CLI.

   ```
   aws imagebuilder cancel-image-creation --image-build-version-
   arn arn:aws:imagebuilder:us-east-1:123456789012:image-pipeline/sample-pipeline
   ```

2. Cambie todos los cronogramas de canalización para usar un proceso de compilación manual o elimínelos si no los va a volver a usar. Para obtener más información sobre cómo eliminas recursos, consulte Eliminación de recursos de Generador de imágenes obsoletos o no utilizados.

Cómo eliminar el rol vinculado a servicios con IAM

Puede utilizar la consola de IAM AWS CLI, la o la AWS API para eliminar el `AWSServiceRoleForImageBuilder` rol de su cuenta. Para obtener más información, consulte Eliminar un rol vinculado a un servicio en la Guía del usuario de IAM.

Regiones admitidas para los roles vinculados a servicios de Generador de imágenes

Image Builder admite el uso de funciones vinculadas a servicios en todas las AWS regiones en las que el servicio esté disponible. Para ver la lista de AWS regiones compatibles, consulte. AWS Regiones y puntos de enlace

Solución de problemas de IAM en Generador de imágenes

Temas

- No tengo autorización para realizar una acción en Generador de Imágenes.
- No estoy autorizado a realizar tareas como: PassRole
- Quiero permitir que personas ajenas a mí accedan Cuenta de AWS a mis recursos de Image Builder

No tengo autorización para realizar una acción en Generador de Imágenes.

Si recibe un error que indica que no tiene autorización para realizar una acción, las políticas se deben actualizar para permitirle realizar la acción.

En el siguiente ejemplo, el error se produce cuando el usuario de IAM `mateojackson` intenta utilizar la consola para consultar los detalles acerca de un recurso ficticio *`my-example-widget`*, pero no tiene los permisos ficticios `imagebuilder:`*`GetWidget`*.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 imagebuilder:GetWidget on resource: my-example-widget
```

En este caso, la política del usuario `mateojackson` debe actualizarse para permitir el acceso al recurso *`my-example-widget`* mediante la acción `imagebuilder:`*`GetWidget`*.

Si necesita ayuda, póngase en contacto con su AWS administrador. El administrador es la persona que le proporcionó las credenciales de inicio de sesión.

No estoy autorizado a realizar tareas como: PassRole

Si recibe un error que indica que no tiene autorización para realizar la acción `iam:PassRole`, se deben actualizar las políticas a fin de permitirle pasar un rol a Generador de Imágenes.

Algunas Servicios de AWS permiten transferir una función existente a ese servicio en lugar de crear una nueva función de servicio o una función vinculada a un servicio. Para ello, debe tener permisos para transferir la función al servicio.

En el siguiente ejemplo, el error se produce cuando un usuario de IAM denominado `marymajor` intenta utilizar la consola para realizar una acción en Generador de Imágenes. Sin embargo, la acción requiere que el servicio cuente con permisos que otorguen un rol de servicio. Mary no tiene permisos para transferir el rol al servicio.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

En este caso, las políticas de Mary se deben actualizar para permitirle realizar la acción `iam:PassRole`.

Si necesita ayuda, póngase en contacto con su AWS administrador. El administrador es la persona que le proporcionó las credenciales de inicio de sesión.

Quiero permitir que personas ajenas a mí accedan Cuenta de AWS a mis recursos de Image Builder

Se puede crear un rol que los usuarios de otras cuentas o las personas externas a la organización puedan utilizar para acceder a sus recursos. Se puede especificar una persona de confianza para que asuma el rol. En el caso de los servicios que admiten políticas basadas en recursos o listas de control de acceso (ACLs), puede utilizar esas políticas para permitir que las personas accedan a sus recursos.

Para obtener más información, consulte lo siguiente:

- Para conocer si Generador de Imágenes admite estas características, consulte Funcionamiento del Generador de imágenes con las políticas y roles de IAM.
- Para obtener información sobre cómo proporcionar acceso a los recursos de su Cuentas de AWS propiedad, consulte Proporcionar acceso a un usuario de IAM en otro de su propiedad en la Cuenta de AWS Guía del usuario de IAM.
- Para obtener información sobre cómo proporcionar acceso a tus recursos a terceros Cuentas de AWS, consulta Cómo proporcionar acceso a recursos que Cuentas de AWS son propiedad de terceros en la Guía del usuario de IAM.

- Para obtener información sobre cómo proporcionar acceso mediante una federación de identidades, consulte Proporcionar acceso a usuarios autenticados externamente (identidad federada) en la Guía del usuario de IAM.
- Para conocer sobre la diferencia entre las políticas basadas en roles y en recursos para el acceso entre cuentas, consulte Acceso a recursos entre cuentas en IAM en la Guía del usuario de IAM.

Recursos de validación de conformidad para Generador de imágenes

EC2 Image Builder no está incluido en el ámbito de ningún programa de AWS conformidad.

Para obtener una lista del alcance Servicios de AWS de los programas de cumplimiento específicos, consulte AWS Servicios dentro del alcance por programa de cumplimiento Servicios de AWS en el . Para obtener información general, consulte Programas de AWS cumplimiento > Programas AWS .

Puede descargar informes de auditoría de terceros utilizando AWS Artifact. Para obtener más información, consulte Descarga de informes en AWS Artifact .

Su responsabilidad en el ámbito de la conformidad al usar Generador de Imágenes viene determinada por la confidencialidad de los datos, los objetivos de conformidad de su empresa y las leyes y regulaciones aplicables.AWS proporciona los siguientes recursos para ayudarlo con la conformidad:

- Security and Compliance Quick Start Guides (Guías de inicio rápido de seguridad y conformidad) (Guías de inicio rápido de seguridad y conformidad): Estas guías de implementación analizan las consideraciones en materia de arquitectura y proporcionan los pasos para implementar los entornos de referencia centrados en la seguridad y la conformidad en AWS.
- AWS Recursos de de cumplimiento: esta colección de libros de trabajo y guías puede aplicarse a su industria y ubicación.
- Evaluación de los recursos con las reglas de la guía para AWS Config desarrolladores: el AWS Config servicio evalúa en qué medida las configuraciones de los recursos cumplen con las prácticas internas, las directrices del sector y las normas.
- AWS Security Hub CSPM— Este AWS servicio proporciona una visión integral del estado de su seguridad AWS que le ayuda a comprobar su conformidad con los estándares y las mejores prácticas del sector de la seguridad.

Puede incorporar productos de conformidad AWS Marketplace o componentes de Ejecutor y orquestador de tareas de AWS(TOE de AWS) en sus imágenes de Image Builder para garantizar que sus imágenes sean conformes. Para obtener más información, consulte Productos de conformidad para sus imágenes de Generador de Imágenes.

Redundancia y resiliencia de datos en Generador de imágenes

La infraestructura AWS global se basa en AWS regiones y zonas de disponibilidad.AWS Las regiones proporcionan varias zonas de disponibilidad aisladas y separadas físicamente, que están conectadas mediante redes de baja latencia, alto rendimiento y alta redundancia. Con las zonas de disponibilidad, puede diseñar y utilizar aplicaciones y bases de datos que realizan una conmutación por error automática entre las zonas sin interrupciones. Las zonas de disponibilidad tienen una mayor disponibilidad, tolerancia a errores y escalabilidad que las infraestructuras tradicionales de uno o varios centros de datos.

El servicio EC2 Image Builder le permite distribuir imágenes creadas en una región con otras regiones, lo que les proporciona resiliencia multirregional. AMIs No existe ningún mecanismo para “hacer copias de seguridad” del flujo, recetas o componentes de imágenes. Puede almacenar los documentos de recetas y componentes fuera del servicio Generador de Imágenes, por ejemplo, en un bucket de Amazon S3.

El EC2 Image Builder no se puede configurar para alta disponibilidad (HA). Puede distribuir imágenes en varias regiones para aumentar la disponibilidad de las imágenes.

Para obtener más información sobre AWS las regiones y las zonas de disponibilidad, consulte Infraestructura AWS global.

Seguridad de la infraestructura en Generador de Imágenes

La red AWS global proporciona capacidades de seguridad y controla el acceso a la red para servicios como EC2 Image Builder. Para obtener más información sobre la seguridad de infraestructura que AWS proporciona a sus servicios, consulte la sección Seguridad de la infraestructura en el documento técnico Introducción a la seguridad de AWS.

Para enviar solicitudes a través de la red AWS global de acciones de la API Image Builder, el software de su cliente debe cumplir con las siguientes pautas de seguridad:

- Para enviar solicitudes de acciones de la API Generador de Imágenes, el software cliente debe usar una versión compatible de seguridad de la capa de transporte (TLS).

> **Note**
>
> AWS está eliminando gradualmente la compatibilidad con las versiones 1.0 y 1.1 de TLS. Le recomendamos encarecidamente que actualice el software cliente para que utilice la versión 1.2 o posterior de TLS, de modo que pueda seguir conectándose. Para obtener más información, consulte esta entrada de blog de seguridad de AWS.

- El software cliente también debe ser compatible con conjuntos de cifrado con confidencialidad directa total (PFS), como Ephemeral Diffie-Hellman (DHE) o Elliptic Curve Ephemeral Diffie-Hellman (ECDHE). La mayoría de los sistemas actuales, como Java 7 y posteriores, son compatibles con estos modos.
- Debe firmar sus solicitudes de API con un identificador de clave de acceso y una clave de acceso secreta que estén asociadas a una entidad principal AWS Identity and Access Management(IAM). También puede utilizar AWS Security Token Service (AWS STS) para generar credenciales de seguridad temporales para sus solicitudes.

Además, debe tener acceso a las EC2 instancias que Image Builder utiliza para crear y probar imágenes AWS Systems Manager.

Administración de parches para imágenes de Generador de imágenes

AWS proporciona actualizaciones gestionadas AMIs cada mes con las últimas actualizaciones y parches de seguridad aplicados a los siguientes sistemas operativos. Puede utilizarlos AMIs como imagen base para sus personalizaciones. Para obtener más información, consulte Sistemas operativos compatibles.

- Distribuciones de Linux, incluidas Amazon Linux 2, AL2 023, Red Hat Enterprise Linux (RHEL), CentOS, Ubuntu y SUSE Linux Enterprise Server
- Windows Server 2016 y versiones posteriores
- macOS 10.14.x y versiones posteriores

Tras crear una imagen personalizada, eres responsable de aplicar los parches al EC2 sistema de Amazon, según el modelo de responsabilidad compartida. Si las EC2 instancias de la carga de

trabajo de la aplicación se pueden reemplazar fácilmente, lo más eficaz sería actualizar la AMI base y volver a implementar todos los nodos de procesamiento en función de esta imagen.

> **Note**
>
> Para aplicar parches a macOS, le recomendamos que cree una nueva versión de la receta que utilice la última AMI administrada para la imagen base y, a continuación, cree una imagen personalizada actualizada a partir de la receta y del resto de los recursos de generación de imágenes. Si tus instancias Mac no se sustituyen fácilmente, consulta la página Actualizar el sistema operativo y el software de las instancias Mac en la Guía del EC2 usuario de Amazon para obtener más información.

Las siguientes son dos formas de mantener actualizado el Image Builder AMIs .

- AWS-componentes de parches proporcionados: EC2 Image Builder proporciona los siguientes componentes de compilación que instalan todas las actualizaciones pendientes del sistema operativo:
  - `update-linux`
  - `update-windows`

  Estos componentes utilizan el módulo de acción `UpdateOS`. Para obtener más información, consulte Actualizar OS. Los componentes se pueden añadir a los procesos de creación de imágenes seleccionándolos de la lista de componentes proporcionados de AWS.
- Componentes de compilación personalizados con operaciones de aplicación de parches: para instalar o actualizar los parches de forma selectiva en los sistemas operativos compatibles AMIs, puede crear un componente de Image Builder para instalar los parches necesarios. Un componente personalizado puede instalar los parches mediante scripts de shell (Bash o PowerShell), o puede usar el módulo de `UpdateOS` acción para especificar los parches para su instalación o exclusión. Para obtener más información, consulte Módulos de acción compatibles con el administrador de componentes TOE de AWS.

  Componente que usa el módulo de acción `UpdateOS` (solo para Linux y Windows. El módulo de acción `UpdateOS` no es compatible con macOS.)

```
schemaVersion: 1.0
phases:
  - name: build
```

```
 steps:
   - name: UpdateOS
  action: UpdateOS
```

Componente que usa Bash para instalar las actualizaciones de yum

```
schemaVersion: 1.0
phases:
  - name: build
 steps:
   - name: InstallYumUpdates
  action: ExecuteBash
  inputs:
    commands:
   - sudo yum update -y
```

Prácticas recomendadas de seguridad para Generador de imágenes

EC2 Image Builder proporciona una serie de funciones de seguridad que debe tener en cuenta a la hora de desarrollar e implementar sus propias políticas de seguridad. Las siguientes prácticas recomendadas son directrices generales y no constituyen una solución de seguridad completa. Puesto que es posible que estas prácticas recomendadas no sean adecuadas o suficientes para el entorno, considérelas como consideraciones útiles en lugar de como normas.

- No utilice grupos de seguridad demasiado permisivos en las recetas de Generador de Imágenes.
- No comparta imágenes con cuentas en las que no confíe.
- No publique imágenes que contengan datos privados o confidenciales.
- Aplique todos los parches de seguridad disponibles para Windows o Linux durante la creación de imágenes.
- Aplica periódicamente actualizaciones de AMI administradas a sus recetas de macOS y genera nuevas imágenes para lanzar instancias que tengan los últimos parches de seguridad.

Le recomendamos encarecidamente que pruebe las imágenes para validar la postura de seguridad y los niveles de cumplimiento de seguridad aplicables. Soluciones como Amazon Inspector pueden ayudar a validar la postura de seguridad y conformidad de las imágenes.

IMDSv2 para canalizaciones de Image Builder

Cuando se ejecuta la canalización de Image Builder, envía solicitudes HTTP para lanzar EC2 instancias que Image Builder utiliza para crear y probar la imagen. Para configurar la versión de IMDS que utiliza su canalización para las solicitudes de lanzamiento, defina el parámetro `httpTokens` en los ajustes de metadatos de la instancia de configuración de infraestructura de Generador de Imágenes.

> **Note**
>
> Le recomendamos que configure todas las EC2 instancias que Image Builder lance desde una compilación de canalización para IMDSv2 que las solicitudes de recuperación de metadatos de las instancias requieran un encabezado de token firmado.

Para obtener más información acerca de la configuración de la infraestructura de Generador de Imágenes, consulte Administre la configuración de la infraestructura de Generador de imágenes. Para obtener más información sobre las opciones de metadatos de EC2 instancia para imágenes de Linux, consulte Configurar las opciones de metadatos de instancia en la Guía del EC2 usuario de Amazon. Para las imágenes de Windows, consulta Configurar las opciones de metadatos de la instancia en la Guía del EC2 usuario de Amazon.

Limpieza posterior a la creación necesaria

Una vez que Generador de Imágenes complete todos los pasos de creación de la imagen personalizada, Generador de Imágenes prepara la instancia de creación para probarla y crear la imagen. Antes de cerrar la instancia de creación para crear la instantánea, Generador de Imágenes realiza la siguiente limpieza para garantizar la seguridad de la imagen:

Linux

La canalización de Generador de Imágenes ejecuta un script de limpieza para garantizar que la imagen final siga las mejores prácticas de seguridad y para eliminar cualquier artefacto de creación o configuración que no deba transferirse a la instantánea. Sin embargo, puede omitir secciones del script o anular por completo los datos del usuario. Por lo tanto, las imágenes producidas por las canalizaciones de Generador de Imágenes no cumplen necesariamente con ningún criterio reglamentario específico.

Cuando la canalización finaliza sus etapas de creación y prueba, Generador de Imágenes ejecuta automáticamente el siguiente script de limpieza justo antes de crear la imagen de salida.

> **Important**
>
> Si anula los datos de usuario en su receta, el script no se ejecutará. En ese caso, asegúrese de incluir un comando en sus datos de usuario que cree un archivo vacío llamado `perform_cleanup`. Generador de Imágenes detecta este archivo y ejecuta el script de limpieza antes de crear la nueva imagen.

```
#!/bin/bash
if [[ ! -f {{workingDirectory}}/perform_cleanup ]]; then
    echo "Skipping cleanup"
    exit 0
else
    sudo rm -f {{workingDirectory}}/perform_cleanup
fi

function cleanup() {
    FILES=("$@")
    for FILE in "${FILES[@]}"; do
        if [[ -f "$FILE" ]]; then
            echo "Deleting $FILE";
            sudo shred -zuf $FILE;
        fi;
        if [[ -f $FILE ]]; then
            echo "Failed to delete '$FILE'. Failing."
            exit 1
        fi;
    done
};


# Clean up for cloud-init files
CLOUD_INIT_FILES=(
    "/etc/sudoers.d/90-cloud-init-users"
    "/etc/locale.conf"
    "/var/log/cloud-init.log"
    "/var/log/cloud-init-output.log"
)
if [[ -f {{workingDirectory}}/skip_cleanup_cloudinit_files ]]; then
```

```
    echo "Skipping cleanup of cloud init files"
else
    echo "Cleaning up cloud init files"
    cleanup "${CLOUD_INIT_FILES[@]}"
    if [[ $( sudo find /var/lib/cloud -type f | sudo wc -l ) -gt 0 ]]; then
        echo "Deleting files within /var/lib/cloud/*"
        sudo find /var/lib/cloud -type f -exec shred -zuf {} \;
    fi;

    if [[ $( sudo ls /var/lib/cloud | sudo wc -l ) -gt 0 ]]; then
        echo "Deleting /var/lib/cloud/*"
        sudo rm -rf /var/lib/cloud/* || true
    fi;
fi;


# Clean up for temporary instance files
INSTANCE_FILES=(
    "/etc/.updated"
    "/etc/aliases.db"
    "/etc/hostname"
    "/var/lib/misc/postfix.aliasesdb-stamp"
    "/var/lib/postfix/master.lock"
    "/var/spool/postfix/pid/master.pid"
    "/var/.updated"
    "/var/cache/yum/x86_64/2/.gpgkeyschecked.yum"
)
if [[ -f {{workingDirectory}}/skip_cleanup_instance_files ]]; then
    echo "Skipping cleanup of instance files"
else
    echo "Cleaning up instance files"
    cleanup "${INSTANCE_FILES[@]}"
fi;


# Clean up for ssh files
SSH_FILES=(
    "/etc/ssh/ssh_host_rsa_key"
    "/etc/ssh/ssh_host_rsa_key.pub"
    "/etc/ssh/ssh_host_ecdsa_key"
    "/etc/ssh/ssh_host_ecdsa_key.pub"
    "/etc/ssh/ssh_host_ed25519_key"
    "/etc/ssh/ssh_host_ed25519_key.pub"
    "/root/.ssh/authorized_keys"
```

```
)
if [[ -f {{workingDirectory}}/skip_cleanup_ssh_files ]]; then
    echo "Skipping cleanup of ssh files"
else
    echo "Cleaning up ssh files"
    cleanup "${SSH_FILES[@]}"
    USERS=$(ls /home/)
    for user in $USERS; do
        echo Deleting /home/"$user"/.ssh/authorized_keys;
        sudo find /home/"$user"/.ssh/authorized_keys -type f -exec shred -zuf {} \;
    done
    for user in $USERS; do
        if [[ -f /home/"$user"/.ssh/authorized_keys ]]; then
            echo Failed to delete /home/"$user"/.ssh/authorized_keys;
            exit 1
        fi;
    done;
fi;


# Clean up for instance log files
INSTANCE_LOG_FILES=(
    "/var/log/audit/audit.log"
    "/var/log/boot.log"
    "/var/log/dmesg"
    "/var/log/cron"
)
if [[ -f {{workingDirectory}}/skip_cleanup_instance_log_files ]]; then
    echo "Skipping cleanup of instance log files"
else
    echo "Cleaning up instance log files"
    cleanup "${INSTANCE_LOG_FILES[@]}"
fi;

# Clean up for TOE files
if [[ -f {{workingDirectory}}/skip_cleanup_toe_files ]]; then
    echo "Skipping cleanup of TOE files"
else
    echo "Cleaning TOE files"
    if [[ $( sudo find {{workingDirectory}}/TOE_* -type f | sudo wc -l) -gt 0 ]];
 then
        echo "Deleting files within {{workingDirectory}}/TOE_*"
        sudo find {{workingDirectory}}/TOE_* -type f -exec shred -zuf {} \;
    fi
```

```
    if [[ $( sudo find {{workingDirectory}}/TOE_* -type f | sudo wc -l) -gt 0 ]];
 then
        echo "Failed to delete {{workingDirectory}}/TOE_*"
        exit 1
    fi
    if [[ $( sudo find {{workingDirectory}}/TOE_* -type d | sudo wc -l) -gt 0 ]];
 then
        echo "Deleting {{workingDirectory}}/TOE_*"
        sudo rm -rf {{workingDirectory}}/TOE_*
    fi
    if [[ $( sudo find {{workingDirectory}}/TOE_* -type d | sudo wc -l) -gt 0 ]];
 then
        echo "Failed to delete {{workingDirectory}}/TOE_*"
        exit 1
    fi
fi

# Clean up for ssm log files
if [[ -f {{workingDirectory}}/skip_cleanup_ssm_log_files ]]; then
    echo "Skipping cleanup of ssm log files"
else
    echo "Cleaning up ssm log files"
    if [[ $( sudo find /var/log/amazon/ssm -type f | sudo wc -l) -gt 0 ]]; then
        echo "Deleting files within /var/log/amazon/ssm/*"
        sudo find /var/log/amazon/ssm -type f -exec shred -zuf {} \;
    fi
    if [[ $( sudo find /var/log/amazon/ssm -type f | sudo wc -l) -gt 0 ]]; then
        echo "Failed to delete /var/log/amazon/ssm"
        exit 1
    fi
    if [[ -d "/var/log/amazon/ssm" ]]; then
        echo "Deleting /var/log/amazon/ssm/*"
        sudo rm -rf /var/log/amazon/ssm
    fi
    if [[ -d "/var/log/amazon/ssm" ]]; then
        echo "Failed to delete /var/log/amazon/ssm"
        exit 1
    fi
fi


if [[ $( sudo find /var/log/sa/sa* -type f | sudo wc -l ) -gt 0 ]]; then
    echo "Deleting /var/log/sa/sa*"
    sudo shred -zuf /var/log/sa/sa*
```

```
fi
if [[ $( sudo find /var/log/sa/sa* -type f | sudo wc -l ) -gt 0 ]]; then
    echo "Failed to delete /var/log/sa/sa*"
    exit 1
fi

if [[ $( sudo find /var/lib/dhclient/dhclient*.lease -type f | sudo wc -l ) -gt
 0 ]]; then
        echo "Deleting /var/lib/dhclient/dhclient*.lease"
        sudo shred -zuf /var/lib/dhclient/dhclient*.lease
fi
if [[ $( sudo find /var/lib/dhclient/dhclient*.lease -type f | sudo wc -l ) -gt
 0 ]]; then
        echo "Failed to delete /var/lib/dhclient/dhclient*.lease"
        exit 1
fi

if [[ $( sudo find /var/tmp -type f | sudo wc -l) -gt 0 ]]; then
        echo "Deleting files within /var/tmp/*"
        sudo find /var/tmp -type f -exec shred -zuf {} \;
fi
if [[ $( sudo find /var/tmp -type f | sudo wc -l) -gt 0 ]]; then
        echo "Failed to delete /var/tmp"
        exit 1
fi
if [[ $( sudo ls /var/tmp | sudo wc -l ) -gt 0 ]]; then
        echo "Deleting /var/tmp/*"
        sudo rm -rf /var/tmp/*
fi

# Shredding is not guaranteed to work well on rolling logs

if [[ -f "/var/lib/rsyslog/imjournal.state" ]]; then
        echo "Deleting /var/lib/rsyslog/imjournal.state"
        sudo shred -zuf /var/lib/rsyslog/imjournal.state
        sudo rm -f /var/lib/rsyslog/imjournal.state
fi

if [[ $( sudo ls /var/log/journal/ | sudo wc -l ) -gt 0 ]]; then
        echo "Deleting /var/log/journal/*"
        sudo find /var/log/journal/ -type f -exec shred -zuf {} \;
        sudo rm -rf /var/log/journal/*
fi
```

```
sudo touch /etc/machine-id
```

Windows

Una vez que la canalización de Generador de imágenes personalice las imágenes de Windows, ejecuta el servicio Sysprep de Microsoft. Estas acciones siguen las prácticas AWS recomendadas para endurecer y limpiar la imagen.

macOS

La canalización de Generador de Imágenes ejecuta un script de limpieza para garantizar que la imagen final siga las mejores prácticas de seguridad y para eliminar cualquier artefacto de creación o configuración que no deba transferirse a la instantánea. Sin embargo, puede omitir secciones del script o anular por completo los datos del usuario. Por lo tanto, las imágenes producidas por las canalizaciones de Generador de Imágenes no cumplen necesariamente con ningún criterio reglamentario específico.

Cuando la canalización finaliza sus etapas de creación y prueba, Generador de Imágenes ejecuta automáticamente el siguiente script de limpieza justo antes de crear la imagen de salida.

> **Important**
>
> Si anula los datos de usuario en su receta, el script no se ejecutará. En ese caso, asegúrese de incluir un comando en sus datos de usuario que cree un archivo vacío llamado `perform_cleanup`. Generador de Imágenes detecta este archivo y ejecuta el script de limpieza antes de crear la nueva imagen.

```
#!/bin/bash
if [[ ! -f {{workingDirectory}}/perform_cleanup ]]; then
  echo "Skipping cleanup"
  exit 0
else
  sudo rm -f {{workingDirectory}}/perform_cleanup
fi

function cleanup() {
  FILES=("$@")
  for FILE in "${FILES[@]}"; do
      if [[ -f "$FILE" ]]; then
          echo "Deleting $FILE";
```

```
          sudo rm -f $FILE;
      fi;
      if [[ -f $FILE ]]; then
          echo "Failed to delete '$FILE'. Failing."
          exit 1
      fi;
  done
};

# Clean up for cloud-init files
CLOUD_INIT_FILES=(
  "/etc/sudoers.d/90-cloud-init-users"
  "/etc/locale.conf"
  "/var/log/cloud-init.log"
  "/var/log/cloud-init-output.log"
)
if [[ -f {{workingDirectory}}/skip_cleanup_cloudinit_files ]]; then
  echo "Skipping cleanup of cloud init files"
else
  echo "Cleaning up cloud init files"
  cleanup "${CLOUD_INIT_FILES[@]}"
  if [[ $( sudo find /var/lib/cloud -type f | sudo wc -l ) -gt 0 ]]; then
      echo "Deleting files within /var/lib/cloud/*"
      sudo find /var/lib/cloud -type f -exec rm -f {} \;
  fi;

  if [[ $( sudo ls /var/lib/cloud | sudo wc -l ) -gt 0 ]]; then
      echo "Deleting /var/lib/cloud/*"
      sudo rm -rf /var/lib/cloud/* || true
  fi;
fi;


# Clean up for temporary instance files
INSTANCE_FILES=(
  "/etc/.updated"
  "/etc/aliases.db"
  "/etc/hostname"
  "/var/lib/misc/postfix.aliasesdb-stamp"
  "/var/lib/postfix/master.lock"
  "/var/spool/postfix/pid/master.pid"
  "/var/.updated"
  "/var/cache/yum/x86_64/2/.gpgkeyschecked.yum"
)
```

```
if [[ -f {{workingDirectory}}/skip_cleanup_instance_files ]]; then
  echo "Skipping cleanup of instance files"
else
  echo "Cleaning up instance files"
  cleanup "${INSTANCE_FILES[@]}"
fi;


# Clean up for ssh files
SSH_FILES=(
  "/etc/ssh/ssh_host_rsa_key"
  "/etc/ssh/ssh_host_rsa_key.pub"
  "/etc/ssh/ssh_host_ecdsa_key"
  "/etc/ssh/ssh_host_ecdsa_key.pub"
  "/etc/ssh/ssh_host_ed25519_key"
  "/etc/ssh/ssh_host_ed25519_key.pub"
  "/root/.ssh/authorized_keys"
)
if [[ -f {{workingDirectory}}/skip_cleanup_ssh_files ]]; then
  echo "Skipping cleanup of ssh files"
else
  echo "Cleaning up ssh files"
  cleanup "${SSH_FILES[@]}"
  USERS=$(ls /home/)
  for user in $USERS; do
      echo Deleting /home/"$user"/.ssh/authorized_keys;
      sudo find /home/"$user"/.ssh/authorized_keys -type f -exec rm -f {} \;
  done
  for user in $USERS; do
      if [[ -f /home/"$user"/.ssh/authorized_keys ]]; then
          echo Failed to delete /home/"$user"/.ssh/authorized_keys;
          exit 1
      fi;
  done;
fi;


# Clean up for instance log files
INSTANCE_LOG_FILES=(
  "/var/log/audit/audit.log"
  "/var/log/boot.log"
  "/var/log/dmesg"
  "/var/log/cron"
  "/var/log/amazon/ec2/ec2-macos-init.log"
```

```
  "/var/log/amazon/ec2/ena-ethernet.log"
  "/var/log/amazon/ec2/system-monitoring.log"
)
if [[ -f {{workingDirectory}}/skip_cleanup_instance_log_files ]]; then
  echo "Skipping cleanup of instance log files"
else
  echo "Cleaning up instance log files"
  cleanup "${INSTANCE_LOG_FILES[@]}"
fi;

# Clean up for TOE files
if [[ -f {{workingDirectory}}/skip_cleanup_toe_files ]]; then
  echo "Skipping cleanup of TOE files"
else
  echo "Cleaning TOE files"
  if [[ $( sudo find {{workingDirectory}}/TOE_* -type f | sudo wc -l) -gt 0 ]]; then
      echo "Deleting files within {{workingDirectory}}/TOE_*"
      sudo find {{workingDirectory}}/TOE_* -type f -exec rm -f {} \;
  fi
  if [[ $( sudo find {{workingDirectory}}/TOE_* -type f | sudo wc -l) -gt 0 ]]; then
      echo "Failed to delete {{workingDirectory}}/TOE_*"
      exit 1
  fi
  if [[ $( sudo find {{workingDirectory}}/TOE_* -type d | sudo wc -l) -gt 0 ]]; then
      echo "Deleting {{workingDirectory}}/TOE_*"
      sudo rm -rf {{workingDirectory}}/TOE_*
  fi
  if [[ $( sudo find {{workingDirectory}}/TOE_* -type d | sudo wc -l) -gt 0 ]]; then
      echo "Failed to delete {{workingDirectory}}/TOE_*"
      exit 1
  fi
fi

# Clean up for ssm log files
if [[ -f {{workingDirectory}}/skip_cleanup_ssm_log_files ]]; then
  echo "Skipping cleanup of ssm log files"
else
  echo "Cleaning up ssm log files"
  if [[ $( sudo find /var/log/amazon/ssm -type f | sudo wc -l) -gt 0 ]]; then
      echo "Deleting files within /var/log/amazon/ssm/*"
      sudo find /var/log/amazon/ssm -type f -exec rm -f {} \;
  fi
  if [[ $( sudo find /var/log/amazon/ssm -type f | sudo wc -l) -gt 0 ]]; then
      echo "Failed to delete /var/log/amazon/ssm"
```

```
      exit 1
  fi
  if [[ -d "/var/log/amazon/ssm" ]]; then
      echo "Deleting /var/log/amazon/ssm/*"
      sudo rm -rf /var/log/amazon/ssm
  fi
  if [[ -d "/var/log/amazon/ssm" ]]; then
      echo "Failed to delete /var/log/amazon/ssm"
      exit 1
  fi
fi


if [[ $( sudo find /var/log/sa/sa* -type f | sudo wc -l ) -gt 0 ]]; then
  echo "Deleting /var/log/sa/sa*"
  sudo rm -f /var/log/sa/sa*
fi
if [[ $( sudo find /var/log/sa/sa* -type f | sudo wc -l ) -gt 0 ]]; then
  echo "Failed to delete /var/log/sa/sa*"
  exit 1
fi

if [[ $( sudo find /var/lib/dhclient/dhclient*.lease -type f | sudo wc -l ) -gt
 0 ]]; then
      echo "Deleting /var/lib/dhclient/dhclient*.lease"
      sudo rm -f /var/lib/dhclient/dhclient*.lease
fi
if [[ $( sudo find /var/lib/dhclient/dhclient*.lease -type f | sudo wc -l ) -gt
 0 ]]; then
      echo "Failed to delete /var/lib/dhclient/dhclient*.lease"
      exit 1
fi

if [[ $( sudo find /var/tmp -type f | sudo wc -l) -gt 0 ]]; then
      echo "Deleting files within /var/tmp/*"
      sudo find /var/tmp -type f -exec rm -f {} \;
fi
if [[ $( sudo find /var/tmp -type f | sudo wc -l) -gt 0 ]]; then
      echo "Failed to delete /var/tmp"
      exit 1
fi
if [[ $( sudo ls /var/tmp | sudo wc -l ) -gt 0 ]]; then
      echo "Deleting /var/tmp/*"
      sudo rm -rf /var/tmp/*
```

```
fi

# Shredding is not guaranteed to work well on rolling logs

if [[ -f "/var/lib/rsyslog/imjournal.state" ]]; then
      echo "Deleting /var/lib/rsyslog/imjournal.state"
      sudo rm -f /var/lib/rsyslog/imjournal.state
      sudo rm -f /var/lib/rsyslog/imjournal.state
fi

if [[ $( sudo ls /var/log/journal/ | sudo wc -l ) -gt 0 ]]; then
      echo "Deleting /var/log/journal/*"
      sudo find /var/log/journal/ -type f -exec rm -f {} \;
      sudo rm -rf /var/log/journal/*
fi

sudo touch /etc/machine-id
```

Anule el script de limpieza de Linux.

Generador de Imágenes crea imágenes que son seguras de forma predeterminada y siguen nuestras prácticas recomendadas de seguridad. Sin embargo, algunos casos de uso más avanzados pueden requerir que omita una o más secciones del script de limpieza integrado. Si necesita omitir parte de la limpieza, le recomendamos encarecidamente que pruebe la AMI de salida para garantizar la seguridad de su imagen.

> **Important**
>
> Si se omiten secciones del script de limpieza, es posible que información confidencial, como los detalles de la cuenta del propietario o las claves SSH, se incluya en la imagen final y, en cualquier instancia que se lance desde esa imagen. También es posible que tenga problemas con el lanzamiento en distintas zonas de disponibilidad, regiones o cuentas.

En la siguiente tabla se describen las secciones del script de limpieza, los archivos que se eliminan en esa sección y los nombres de archivo que puede utilizar para marcar una sección que Generador de Imágenes debe omitir. Para omitir una sección específica del script de limpieza, puede utilizar el módulo de acción del componente CreateFile o un comando de los datos de usuario (si es incorrecto)

para crear un archivo vacío con el nombre especificado en la columna Omitir el nombre del archivo de la sección.

> **Note**
>
> Los archivos que cree para omitir una sección del script de limpieza no deben incluir una extensión del archivo. Por ejemplo, si desea omitir la sección `CLOUD_INIT_FILES` del script, pero crea un archivo denominado `skip_cleanup_cloudinit_files.txt`, Generador de Imágenes no reconocerá el archivo omitido.

Input

Sección de limpieza	Archivos eliminados	Omita el nombre del archivo de la sección.

`CLOUD_INIT_FILES`	`/etc/sudoers.d/90-cloud-init-users`
	`/etc/locale.conf`
	`/var/log/cloud-init.log`
	`/var/log/cloud-init-output.log`
		`skip_cleanup_cloudinit_files`

`INSTANCE_FILES`	`/etc/.updated`
	`/etc/aliases.db`
	`/etc/hostname`
	`/var/lib/misc/postfix.aliasesdb-stamp`
	`/var/lib/postfix/master.lock`
	`/var/spool/postfix/pid/master.pid`
		`skip_cleanup_instance_files`

Sección de limpieza	Archivos eliminados	Omita el nombre del archivo de la sección.
	/var/.updated /var/cache/yum/x86_64/2/.gpgkeyschecked.yum	
SSH_FILES	/etc/ssh/ssh_host_rsa_key /etc/ssh/ssh_host_rsa_key.pub /etc/ssh/ssh_host_ecdsa_key /etc/ssh/ssh_host_ecdsa_key.pub /etc/ssh/ssh_host_ed25519_key /etc/ssh/ssh_host_ed25519_key.pub /root/.ssh/authorized_keys /home/<all users>/.ssh/authorized_keys;	skip_cleanup_ssh_files

Sección de limpieza	Archivos eliminados	Omita el nombre del archivo de la sección.
INSTANCE_LOG_FILES	/var/log/audit/audit.log /var/log/boot.log /var/log/dmesg /var/log/cron	skip_cleanup_instance_log_files
TOE_FILES	{{workingDirectory}}/TOE_*	skip_cleanup_toe_files
SSM_LOG_FILES	/var/log/amazon/ssm/*	skip_cleanup_ssm_log_files

Solucionar problemas de Generador de imágenes

EC2 Image Builder se integra con Servicios de AWS el monitoreo y la solución de problemas para ayudarlo a solucionar problemas de creación de imágenes. Generador de Imágenes registra y muestra el progreso de cada paso del proceso de creación de imágenes. Además, Generador de Imágenes puede exportar registros a la ubicación de Amazon S3 que usted proporcione.

Para la solución avanzada de problemas, puede ejecutar comandos y scripts predefinidos mediante Ejecución de comandos de AWS Systems Manager.

Contenido

- Solución de problemas de canalizaciones
- Escenarios de solución de problemas

Solución de problemas de canalizaciones

Si se produce un error en la compilación de una canalización de Generador de Imágenes, este devuelve un mensaje de error que describe el error. Generador de Imágenes también devuelve una `workflow execution ID` en el mensaje de error, como en el siguiente ejemplo de resultado:

```
Workflow Execution ID: wf-12345abc-6789-0123-abc4-567890123abc failed with reason: …
```

Generador de Imágenes organiza y dirige las acciones de creación de imágenes mediante una serie de pasos que se definen para las etapas de tiempo de ejecución de su proceso de creación de imágenes estándar. Cada una de las etapas de creación y prueba del proceso tiene un flujo de trabajo asociado. Cuando Generador de Imágenes ejecuta un flujo de trabajo para crear o probar una nueva imagen, genera un recurso de metadatos del flujo de trabajo que realiza un seguimiento de los detalles del tiempo de ejecución.

Las imágenes de contenedor tienen un flujo de trabajo adicional que se ejecuta durante la distribución.

Investigación de los detalles de los errores de las instancias de tiempo de ejecución de su flujo de trabajo

Para solucionar un error en el tiempo de ejecución de tu flujo de trabajo, puedes llamar a las acciones GetWorkflowExecutiony a la ListWorkflowStepExecutionsAPI con tu. `workflow execution ID`

Revisión de los registros de tiempo de ejecución del flujo de trabajo

- Amazon CloudWatch Logs

  Image Builder publica registros detallados de ejecución del flujo de trabajo en el siguiente grupo y flujo de CloudWatch registros de Image Builder:

  Con CloudWatch los registros, puede buscar datos de registro con patrones de filtro. Para obtener más información, consulta Buscar datos de registro mediante patrones de filtro en la Guía del usuario de Amazon CloudWatch Logs.

- AWS CloudTrail

  Todas las actividades de creación también se registran CloudTrail si están activadas en tu cuenta. Puedes filtrar CloudTrail los eventos por fuente`imagebuilder.amazonaws.com`. Como alternativa, puedes buscar el ID de EC2 instancia de Amazon que se devuelve en el registro de ejecución para ver más detalles sobre la ejecución de la canalización.

- Amazon Simple Storage Service (S3)

  Si especificó un nombre de bucket de S3 y un prefijo de clave en la configuración de la infraestructura, la ruta del registro del tiempo de ejecución de los pasos del flujo de trabajo sigue este patrón:

  ```
  S3://S3BucketName/KeyPrefix/ImageName/ImageVersion/ImageBuildVersion/
  WorkflowExecutionId/StepName
  ```

  Los registros que envías a tu bucket de S3 muestran los pasos y los mensajes de error relacionados con la actividad de la EC2 instancia durante el proceso de creación de la imagen. Los registros incluyen los resultados del registro del administrador de componentes, las definiciones de los componentes que se ejecutaron y el resultado detallado (en JSON) de todos los pasos realizados en la instancia. Si encuentra un problema, debería revisar estos archivos, empezando por `application.log`, para diagnosticar la causa del problema en la instancia.

De forma predeterminada, Image Builder cierra la instancia de EC2 compilación o prueba de Amazon que se está ejecutando cuando se produce un error en la canalización. Puede cambiar la configuración de la instancia del recurso de configuración de infraestructura que utiliza la canalización para retener la instancia de compilación o prueba a fin de solucionar problemas.

Para cambiar la configuración de la instancia en la consola, debe desactivar la casilla de verificación Terminar la instancia en caso de fallo, que se encuentra en la sección Configuración de solución de problemas del recurso de configuración de la infraestructura.

También puede cambiar la configuración de la instancia con el comando update-infrastructure-configuration en la AWS CLI. Defina el valor `terminateInstanceOnFailure` como `false` en el archivo JSON al que hace referencia el comando con el parámetro `--cli-input-json`. Para obtener más información, consulte Actualización de una configuración de infraestructura.

Escenarios de solución de problemas

En esta sección se enumeran los siguientes escenarios detallados de solución de problemas:

- Acceso denegado: código de estado 403
- Se agota el tiempo de espera de la compilación al verificar la disponibilidad de Systems Manager Agent en la instancia de compilación.
- El disco secundario de Windows está fuera de línea durante el lanzamiento.
- La compilación falla con la imagen base reforzada de CIS.
- AssertInventoryCollection falla (Automatización de Systems Manager)

Para ver los detalles de un escenario, elija el título del escenario para ampliarlo. Puede tener varios títulos ampliados al mismo tiempo.

Acceso denegado: código de estado 403

Descripción

Se produce un error en la compilación de la canalización con el código de estado «AccessDeniedAcceso denegado: 403".

Causa

Entre las causas posibles se incluyen las siguientes:

- El perfil de la instancia no tiene los permisos necesarios para acceder a los recursos APIs o componentes.
- Al rol del perfil de instancia le faltan los permisos necesarios para el registro en Amazon S3. Por lo general, esto ocurre cuando el rol del perfil de la instancia no tiene PutObjectpermisos para los buckets de S3.

Solución

Según la posible causa, este problema puede resolverse de la siguiente manera:

- Al perfil de instancia le faltan políticas administradas: agregue las políticas que faltan a su rol de perfil de instancia. A continuación, vuelva a ejecutar la canalización.
- Al perfil de instancia le faltan permisos de escritura para el bucket de S3: añada una política a su rol de perfil de instancia que otorgue PutObjectpermisos de escritura en su bucket de S3. A continuación, vuelva a ejecutar la canalización.

Se agota el tiempo de espera de la compilación al verificar la disponibilidad de Systems Manager Agent en la instancia de compilación.

Descripción

La creación de la canalización falla con «status = 'TimedOut'» y «mensaje de error = 'Se agotó el tiempo de espera del paso mientras se verifica la disponibilidad del agente de Systems Manager en las instancias de destino'».

Causa

Entre las causas posibles se incluyen las siguientes:

- La instancia que se lanzó para realizar las operaciones de compilación y ejecutar los componentes no pudo acceder al punto de conexión de Systems Manager.
- El perfil de instancia no cuenta con los permisos requeridos.

Solución

Según la posible causa, este problema se puede resolver de la siguiente manera:

- Problema de acceso, subred privada: si está creando una subred privada, asegúrese de haber configurado PrivateLink puntos de conexión para Systems Manager, Image Builder y, si quiere registrar, Amazon S3/. CloudWatch Para obtener más información sobre la configuración de PrivateLink puntos de conexión, consulte Access services through. AWSAWS PrivateLink
- Permisos faltantes: agregue las siguientes políticas administradas a su rol vinculado al servicio de IAM para Generador de Imágenes:
  - EC2InstanceProfileForImageBuilder

- EC2InstanceProfileForImageBuilderECRContainerCompila
- Amazon SSMManaged InstanceCore

Para obtener más información sobre el rol vinculado a servicios de Generador de Imágenes, consulte Uso de roles vinculados a servicios de IAM para Generador de imágenes.

El disco secundario de Windows está fuera de línea durante el lanzamiento.

Descripción

Si el tipo de instancia utilizado para crear una AMI de Windows de Generador de Imágenes no coincide con el tipo de instancia que se utiliza para lanzar desde la AMI, puede ocurrir un problema cuando los volúmenes no raíz están fuera de línea durante el lanzamiento. Esto ocurre principalmente cuando la instancia de compilación utiliza una arquitectura más nueva que la instancia de lanzamiento.

El siguiente ejemplo muestra lo que ocurre cuando una AMI de Image Builder se crea en un tipo de instancia EC2 Nitro y se lanza en una instancia de EC2 Xen:

Tipo de instancia de compilación: m5.large (Nitro)

Tipo de instancia de lanzamiento: t2.medium (Xen)

```
PS C:\Users\Administrator> get-disk
Number  Friendly Name  Serial Number         Health Status  Operational Status  Total
 Size  Partition Style
------  -------------  -------------         -------------  ------------------
 ----------  ---------------
0       AWS PVDISK     vol0abc12d34e567f8a9  Healthy        Online                  30
 GB  MBR
1       AWS PVDISK     vol1bcd23e45f678a9b0  Healthy        Offline                  8
 GB  MBR
```

Causa

Debido a la configuración predeterminada de Windows, los discos recién descubiertos no se ponen en línea ni se formatean automáticamente. Cuando se activa el tipo de instancia EC2, Windows considera que se están descubriendo nuevos discos. Esto se debe al cambio de controlador subyacente.

Solución

Le recomendamos que utilice el mismo sistema de tipos de instancias al compilar la AMI de Windows desde la que desea iniciar. No incluya tipos de instancias que estén creados en sistemas diferentes en su configuración de infraestructura. Si alguno de los tipos de instancias que especifica usa el sistema Nitro, todos deben usar el sistema Nitro.

Para obtener más información sobre las instancias que se crean en el sistema Nitro, consulta Instancias creadas en el sistema Nitro en la Guía del EC2 usuario de Amazon.

La compilación falla con la imagen base reforzada de CIS.

Descripción

Está utilizando una imagen base reforzada de CIS y la compilación falla.

Causa

Si el directorio `/tmp` se clasifica como`noexec`, se puede producir un error en Generador de Imágenes.

Solución

Elija una ubicación diferente para su directorio de trabajo en el campo `workingDirectory` de la receta de imagen. Para obtener más información, consulte la descripción del tipo ImageRecipede datos.

AssertInventoryCollection falla (Automatización de Systems Manager)

Descripción

Systems Manager Automation muestra un error en el paso de automatización `AssertInventoryCollection`.

Causa

Es posible que usted o su organización hayan creado una asociación de administradores estatales de Systems Manager que recopila información de inventario para EC2 las instancias. Si la recopilación mejorada de metadatos de imágenes está habilitada para la canalización de Generador de Imágenes (esta es la opción predeterminada), Generador de Imágenes intentará crear una nueva asociación de inventario para la instancia de compilación. Sin embargo, Systems Manager no

permite múltiples asociaciones de inventario para las instancias administradas e impide una nueva asociación si ya existe una. Esto produce un error en la operación y provoca un error en la creación de la canalización.

Solución

Para resolver este problema, desactive la recopilación mejorada de metadatos de imágenes mediante uno de los métodos siguientes:

- Actualice la canalización de imágenes en la consola para desactivar la casilla de verificación Habilitar la recopilación mejorada de metadatos. Guarde los cambios y ejecute una compilación de canalización.

  Para obtener más información sobre cómo actualizar la canalización de imágenes de AMI mediante la consola EC2 Image Builder, consulteActualización de las canalizaciones de imágenes de AMI desde la consola. Para obtener más información sobre cómo actualizar la canalización de imágenes de contenedores mediante la consola EC2 Image Builder, consulteActualización de una canalización de imágenes de contenedor desde la consola.
- También puede actualizar la canalización de imágenes con el comando update-image-pipeline en la AWS CLI. Para ello, incluya la propiedad `EnhancedImageMetadataEnabled` en su archivo JSON, establecida como `false`. En el siguiente ejemplo, se muestra la propiedad establecida como `false`.

```
{
    "name": "MyWindows2019Pipeline",
    "description": "Builds Windows 2019 Images",
    "enhancedImageMetadataEnabled": false,
    "imageRecipeArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/my-example-recipe/2020.12.03",
    "infrastructureConfigurationArn": "arn:aws:imagebuilder:us-west-2:123456789012:infrastructure-configuration/my-example-infrastructure-configuration",
    "distributionConfigurationArn": "arn:aws:imagebuilder:us-west-2:123456789012:distribution-configuration/my-example-distribution-configuration",
    "imageTestsConfiguration": {
        "imageTestsEnabled": true,
        "timeoutMinutes": 60
    },
    "schedule": {
        "scheduleExpression": "cron(0 0 * * SUN *)",
```

```
        "pipelineExecutionStartCondition":
 "EXPRESSION_MATCH_AND_DEPENDENCY_UPDATES_AVAILABLE"
    },
    "status": "ENABLED"
}
```

Para evitar que esto suceda con las nuevas canalizaciones, desactive la casilla Habilitar la recopilación mejorada de metadatos al crear una nueva canalización mediante la consola de EC2 Image Builder, o establezca el valor de la `EnhancedImageMetadataEnabled` propiedad en su archivo JSON para `false` cuando cree la AWS CLI canalización mediante.

Historial de documentación de los cambios en la guía del usuario de Generador de imágenes

En la siguiente tabla, se describen los cambios importantes que se han realizado en la documentación por fecha. Para recibir notificaciones sobre los cambios en esta documentación, puede suscribirse a una fuente RSS.

- Versión de la API: 30 de abril de 2020

Cambio	Descripción	Fecha
Mejoras en la distribución de imágenes	Las mejoras incluyen la capacidad de realizar directamente las actividades de distribución sin necesidad de volver a ejecutar una creación de imagen completa.	20 de noviembre de 2025
Mejoras en el control de versiones automático y en el iAC	El incremento automático de versiones y la referencia de versiones con caractere s comodín ahora eliminan la administración manual de versiones para recetas, componentes y flujos de trabajo. Experiencia en pruebas de componentes con ejecución en seco y capacidad de creación mejorada en la consola.	20 de noviembre de 2025
El flujo de trabajo de imágenes es compatible con Lambda y Step Functions	El flujo de trabajo de imágenes ahora admite la invocació n de funciones Lambda y la ejecución de la máquina de estados Step Functions, lo	13 de noviembre de 2025

	que permite flujos de trabajo complejos de varios pasos y una lógica de validación personalizada en la creación de imágenes.	
Mejoras en el procesamiento de imágenes	Las mejoras incluyen la compatibilidad con los registros de ejecución de canalizaciones, CloudWatch los grupos de registros configurables y la configuración para deshabilitar automáticamente las canalizaciones programadas que fallan repetidamente. Las actualizaciones adicionales incluyen información ampliada sobre la programación de canalizaciones y ejemplos de mensajes de eventos de CloudWatch Logs actualizados.	29 de septiembre de 2025
Nuevo script para Amazon Linux 2023	Se agregó soporte para el sistema operativo SUSE Linux Enterprise Server (SLES). Versiones de STIG y STIGS aplicado para la versión del tercer trimestre de 2025 para todos los niveles de conformidad (). low/medium/high	4 de septiembre de 2025
Nuevo script para Amazon Linux 2023	Se agregó un nuevo script diseñado específicamente para Amazon Linux 2023.	26 de agosto de 2025

Actualización de la política de IAM: política de funciones de servicio	Se actualizó la política de funciones vinculadas al servicio para permitir el uso de los parámetros SSM en las recetas y durante la distribución de imágenes. Para obtener más información, consulte la política de AWSServiceRoleForImageBuilder.	23 de julio de 2025
Actualizaciones de STIG para el segundo trimestre de 2025	Se actualizaron las versiones STIG de Windows y se aplicó el STIGS para la versión del segundo trimestre de 2025.	26 de junio de 2025
Actualización de la política de IAM: política de perfil de instancia	Añada `.iso` y `.Iso` como extensiones de archivo válidas para descargar imágenes de disco ISO de Amazon S3. Para obtener más información, consulte la política de EC2InstanceProfileForImageBuilder.	19 de mayo de 2025
Actualizaciones de STIG para el primer trimestre de 2025	Se actualizaron las versiones STIG de Windows y se aplicaron las versiones de los componentes STIGS y SCAP para su versión del primer trimestre de 2025.	5 de mayo de 2025
Versión de función: Support SSM Parameters	Image Builder ahora admite el uso de parámetros del almacén de parámetros AWS Systems Manager(SSM) en recetas y durante la distribución de imágenes.	30 de abril de 2025

Actualizaciones de STIG del cuarto trimestre	Se actualizaron las versiones STIG de Windows y se aplicó el STIGS para la versión del cuarto trimestre de 2024.	4 de febrero de 2025
Actualización de la política de IAM: política de funciones de servicio	Se actualizó la política de funciones vinculadas a servicios para admitir la creación de imágenes a partir de archivos ISO del sistema operativo cliente de Microsoft oficiales importados. Para obtener más información, consulte la política de AWSServiceRoleForImageBuilder.	30 de diciembre de 2024
Actualización de la política de IAM: política de perfil de instancia	Se actualizó la política de roles del perfil de instancia para permitir la creación de imágenes a partir de archivos de imagen de disco. Para obtener más información, consulte la política de EC2InstanceProfileForImageBuilder.	30 de diciembre de 2024
Lanzamiento de funciones: ISO a AMI	Image Builder ahora puede crear una imagen a partir de archivos de disco ISO oficiales para Microsoft Windows 11 y sistemas operativos cliente posteriores.	30 de diciembre de 2024

Actualizaciones de STIG del cuarto trimestre	Se actualizaron las versiones de STIG de Linux y se aplicó el STIGS para su versión del cuarto trimestre de 2024. Se agregó información sobre dos nuevos parámetros de entrada para los componentes de Linux.	10 de diciembre de 2024
Actualización de la política de IAM: política de perfil de instancia	Se actualizó la política del perfil de la instancia para permitir el acceso a AWS Marketplace los componentes. Para obtener más informaci ón, consulte la política de EC2InstanceProfileForImageB uilder.	2 de diciembre de 2024
Lanzamiento de funciones: AWS Marketplace component es	Se agregó soporte para componentes AWS Marketpla ce de software.	2 de diciembre de 2024
Versión de función: compatibi lidad con macOS	Se ha añadido soporte para imágenes de macOS.	22 de octubre de 2024
Soporte documental para operadores lógicos en Ejecutor y orquestador de tareas de AWS	Utilice operadores lógicos para añadir o modificar expresiones condicionales en el documento de componente.	16 de agosto de 2024
Documente el soporte para construcciones condicionales en Ejecutor y orquestador de tareas de AWS	Utilice constructos condicion ales, como la instrucción “if”, para dirigir el flujo de control de las acciones de los componentes en el documento de componentes.	16 de agosto de 2024

Soporte documental para operadores de comparación en Ejecutor y orquestador de tareas de AWS	Utilice los operadores de comparación para comparar los valores del documento de componentes.	16 de agosto de 2024
Se ha agregado el módulo de acción Assert	Se agregó documentación para el módulo acción `ExecuteDocument` en `General execution` .	14 de agosto de 2024
Compatibilidad con sistemas operativos	Se ha agregado compatibilidad con RHEL 9 y Ubuntu 24.04 LTS.	2 de agosto de 2024
Actualización del documento: reorganización del contenido	Reorganización de la documentación para mejorar la presentación y la navegación.	21 de junio de 2024
Actualizaciones de STIG del segundo trimestre	Se han actualizado las versiones de Linux STIG y se ha aplicado el STIGS para la versión del segundo trimestre de 2024. Se ha agregado soporte para RHEL 9, CentOS Stream 9 y Ubuntu 22.04. No se han realizado cambios en las versiones de Windows.	10 de mayo de 2024
Actualizaciones de STIG del primer trimestre	Se han actualizado las versiones de Linux STIG y se aplicó el STIGS para la versión del primer trimestre de 2024. No se han realizado cambios en las versiones de Windows.	23 de febrero de 2024

Actualizaciones de STIG del primer trimestre	Se han actualizado las versiones de Linux STIG y se aplicó el STIGS para la versión del primer trimestre de 2024. No se han realizado cambios en las versiones de Windows.	23 de febrero de 2024
Publicación de características: administración del flujo de trabajo de imágenes	Con los flujos de trabajo de imágenes, tiene más flexibili dad, visibilidad y control sobre el proceso de creación de imágenes. Puede personali zar los pasos de creación y prueba para los flujos de trabajo o puede utilizar el flujo de trabajo predeterminado de Generador de imágenes.	12 de diciembre de 2023
Actualizaciones de STIG del cuarto trimestre	Se han actualizado las versiones de Linux STIG y se ha aplicado el STIGS para la versión del cuarto trimestre de 2023. No se han realizado cambios en las versiones de Windows. También se ha actualizado el SCAP de Linux y Windows para incluir nuevos componentes, software y números de referencia.	7 de diciembre de 2023

Publicación de características: administración del ciclo de vida de imágenes	Con las políticas y reglas de administración del ciclo de vida de imágenes, puede definir su estrategia de administración de recursos para garantizar que las imágenes desactualizadas y sus recursos asociados pasen por un proceso de etiquetado y eliminación.	17 de noviembre de 2023
Actualización de la política de IAM: rol de servicio	Se ha actualizado la nueva política de roles vinculados al servicio para la compatibi lidad con la ubicación de instancias. Para obtener más información, consulte la política de AWSServic eRoleForImageBuilder.	19 de octubre de 2023
Actualizaciones de STIG del tercer trimestre	Se han actualizado las versiones de STIG y se ha aplicado el STIGS para la versión del tercer trimestre de 2023. Mensajes actualiza dos adicionalmente para aclarar que los paquetes de terceros no se instalan automáticamente, con muy pocas excepciones. STIGs Se registran todos los omitidos.	5 de octubre de 2023
Nuevas versiones de STIG	Se han actualizado las versiones de STIG y se ha aplicado el STIGS para la versión del segundo trimestre de 2023.	3 de mayo de 2023

Nuevas versiones de STIG	Se han actualizado las versiones de STIG y se ha aplicado el STIGS para la versión del primer trimestre de 2023. Se agregó soporte para AL2 023.	14 de abril de 2023
Actualice las regiones compatibles para TOE de AWS	Se agregó TOE de AWS soporte para lo siguiente Regiones de AWS: Asia Pacífico (Hyderabad), Asia Pacífico (Yakarta), Europa (Zúrich), Europa (España) y Medio Oriente (Emiratos Árabes Unidos).	13 de abril de 2023
TOE de AWS actualizaciones de descarga de aplicaciones	Se actualizó la firma para la descarga TOE de AWS de la instalación en Windows. El TLS actualizado también indica que para descargar aplicaciones desde buckets de S3 ahora se requiere versión 1.2 o posterior de TLS.	31 de marzo de 2023
Lanzamiento de la caracterí stica: flujos de trabajo de creación mejorados	Se agregaron detalles del tiempo de ejecución de las compilaciones de imágenes en la nueva pestaña de flujo de trabajo en los detalles de la versión de compilaci ón de imágenes. Informaci ón mejorada para solucionar problemas de compilaciones.	30 de marzo de 2023

Lanzamiento de la característica: detección e informes de CVE	En el caso de las cuentas que han activado los análisis de Amazon Inspector, Generador de imágenes puede capturar los resultados de vulnerabilidades y exposiciones comunes (CVE) de Amazon Inspector durante la etapa de prueba del proceso de creación de nuevas imágenes, incluidas las imágenes de contenedores almacenadas en Amazon ECR. Generador de Imágenes crea una instantánea de los resultados para respaldar el análisis detallado. Generador de Imágenes también informa sobre los recuentos de resultados que se pueden filtrar por cuenta, canalización o imagen, con la posibilidad de profundizar en los detalles.	30 de marzo de 2023
Historial de versiones agregado.	Se agregó el historial de versiones a las secciones de Windows y Linux.	17 de febrero de 2023
Nuevas versiones de STIG	Se han actualizado las versiones de STIG y se ha aplicado el STIGS para la versión del cuarto trimestre de 2022.	1 de febrero de 2023

Lanzamiento de funciones: AWS Marketplace integración y endurecimiento del CIS	Se agregó AWS Marketplace la integración para encontrar y usar fácilmente una imagen suscrita como base para una nueva imagen personalizada, incluidas las imágenes reforzadas de CIS y un nuevo componente de CIS Hardening del Centro de Seguridad de Internet.	13 de enero de 2023
Componentes de endurecimiento de CIS	Se agregaron componentes de endurecimiento de CIS que pertenecen a CIS y que son mantenidos por CIS.	13 de enero de 2023
Nuevas versiones de STIG	Se introdujo la compatibilidad con Ubuntu, se actualizaron las versiones de STIG y se aplicó el STIGS para la versión del segundo trimestre de 2022.	20 de julio de 2022
Actualización del documento: navegación para acceder a la página Crear un documento de componente YAML	Se trasladó el contenido del documento del componente Crear componente YAML a su propia página y se actualizaron otras páginas para hacer referencia a él.	7 de junio de 2022
Nuevas versiones de STIG	Se actualizaron las versiones de STIG y se aplicó el STIGS para la versión del primer trimestre de 2022.	25 de abril de 2022

Se agregó un módulo de acción ExecuteDocument	Se agregó documentación para el módulo acción `ExecuteDocument` en `General execution` .	28 de marzo de 2022
Lanzamiento de la característica: apoyo para un inicio más rápido de la AMI de Windows	Se agregaron ajustes de configuración de distribución para permitir un inicio más rápido de Windows AMIs.	21 de febrero de 2022
Versión de mantenimiento: actualice la huella digital TOE de AWS binaria	Se ha actualizado la huella digital binaria del certificado de TOE de AWS firmante.	18 de febrero de 2022
Versión de función: configure la entrada para TOE de AWS	Se agregó soporte para usar un archivo de configuración JSON como entrada para el TOE de AWS run comando.	3 de febrero de 2022
Nuevas versiones de STIG	Se actualizaron las versiones de STIG y se aplicó el STIGS para la versión del cuarto trimestre de 2021. También se agregó una sección para los nuevos componentes de SCAP Compliance Checker (SCC).	22 de diciembre de 2021
Versión de la función: integración de máquinas virtuales Import/Export (VMIE)	Se agregó soporte para la importación de máquinas virtuales a través de todos los canales (consola,API/CLI, etc.), and for VM export via API/CLI. VM export no está disponible actualmente en la consola de Generador de Imágenes.	20 de diciembre de 2021

Lanzamiento de la función: uso compartido de AMI para AWS Organizations y OUs	Se actualizó la configuración de distribución para añadir soporte para compartir la salida AMIs con AWS Organizations y OUs.	24 de noviembre de 2021
Actualización del documento: actualización de las etapas y fases de los componentes	Contenido ampliado para las etapas de los componentes en Image Builder y cómo interactúan con las fases de los TOE de AWS componentes.	22 de septiembre de 2021
Actualización del documento: agregue contenido CloudTrail de integración	Se agregó el resumen de la supervisión y el contenido de CloudTrail integración.	17 de septiembre de 2021
Nuevas versiones de STIG	Se actualizaron las versiones de STIG y se aplicó el STIGS para la versión del tercer trimestre de 2021.	10 de septiembre de 2021
Versión de la función: EventBridge integración con Amazon	Se agregó EventBridge soporte que le permite conectar Image Builder con eventos relacionados Servicios de AWS e iniciar eventos en función de las reglas definidas en EventBridge.	18 de agosto de 2021
Actualización del documento: reordenar las páginas TOE de AWS	TOE de AWS Páginas reorganizadas para mayor claridad.	11 de agosto de 2021

Lanzamiento de la característica: componentes parametrizados y configuración de instancias adicional	Se agregó soporte para especificar parámetros a fin de personalizar los componentes de las recetas. Configuración ampliada de las EC2 instancias que se utilizan para crear y probar imágenes, incluida la capacidad de especificar comandos para que se ejecuten en el lanzamiento y un mayor control sobre la instalación y desinstalación del agente de Systems Manager.	7 de julio de 2021
Nuevas versiones de STIG	Se actualizaron las versiones de STIG y se aplicó el STIGS para la versión del segundo trimestre de 2021.	30 de junio de 2021
Mejora: mejoras en el etiquetado	Se mejoró la mensajería relacionada con el etiquetado de recursos.	25 de junio de 2021
Lanzamiento de la característica: integración de plantillas de lanzamiento	Se ha añadido compatibilidad con el uso de plantillas de EC2 lanzamiento de Amazon para la distribución de AMI en la configuración de distribución.	7 de abril de 2021
Lanzamiento de la característica: mejoras en la creación de contenedores	Se agregó soporte para configurar las asignaciones de dispositivos de bloques y especificar AMIs su uso como imagen base para la creación de contenedores.	7 de abril de 2021

Nuevas versiones de STIG	Versiones actualizadas de STIG y STIG aplicadas.	5 de marzo de 2021
Actualización de expresiones cron	El procesamiento cron de Generador de Imágenes se actualiza para aumentar la granularidad de las expresion es cron al minuto y utiliza un motor de programación cron estándar. Los ejemplos se actualizan con el nuevo formato.	8 de febrero de 2021
Lanzamiento de la caracterí stica: soporte para contenedo res	Se agregó soporte para crear imágenes de contenedores de Docker mediante Generador de Imágenes, con registro y almacenamiento de las imágenes resultantes en Amazon Elastic Container Registry (Amazon ECR). El contenido se ha reorganiz ado para reflejar las nuevas funciones y adaptarse al crecimiento futuro.	17 de diciembre de 2020

Documentación cron reestructurada	Esta página ahora incluye más información sobre cómo funciona cron con las compilaciones en proceso de Generador de Imágenes e incluye detalles sobre la hora UTC. Se han eliminado los caracteres comodín que no estaban permitidos en campos específicos. Los ejemplos ahora incluyen muestras de expresiones para la consola y la CLI.	13 de noviembre de 2020
Consola, versión 2.0: edición de canalizaciones actualizada	El contenido cambia en los tutoriales de introducción y creación de canalizaciones, además de la página de gestión de canalizaciones de imágenes, para incorporar nuevas características y flujos de la consola.	13 de noviembre de 2020
Nuevas versiones de STIG	Versiones actualizadas de STIG y STIG aplicadas. Nota: se ha cambiado el formato de la lista para mostrar los STIGs que se aplican de forma predeterminada.	15 de octubre de 2020
Support para construcciones en bucle en TOE de AWS	Cree constructos en bucle para definir una secuencia repetida de instrucciones en la aplicación TOE de AWS .	29 de julio de 2020

Support for local development of TOE de AWS components	Desarrolle y pruebe los componentes de la imagen localmente con la TOE de AWS aplicación.	28 de julio de 2020
Encriptado AMIs	EC2 Image Builder añade compatibilidad con la distribución cifrada de AMI.	1 de julio de 2020
AutoScaling obsolescencia	Depresión del uso de AutoScaling para lanzar instancias.	15 de junio de 2020
Support para la conectividad a través de AWS PrivateLink	Puede establecer una conexión privada entre su VPC e EC2 Image Builder mediante la creación de un punto final de interfaz de VPC. Los puntos finales de la interfaz cuentan con una tecnología que le permite acceder de forma privada a Image Builder APIs sin una puerta de enlace a Internet, un dispositivo NAT, una conexión VPN o una conexión AWS Direct Connect. AWS PrivateLink Las instancias de su VPC no necesitan direcciones IP públicas para comunicarse con Image Builder. APIs El tráfico entre su VPC e Generador de Imágenes no sale de la red de Amazon.	10 de junio de 2020
Nuevas versiones de STIG	Versiones actualizadas de STIG y STIG aplicadas.	23 de enero de 2020

Solución de problemas	Se agregaron escenarios generales de solución de problemas.	22 de enero de 2020
Componentes STIG	Puede crear imágenes compatibles con STIG con los componentes de STIG. TOE de AWS	22 de enero de 2020

Las traducciones son generadas a través de traducción automática. En caso de conflicto entre la traducción y la version original de inglés, prevalecerá la version en inglés.