aws

Guide de l’utilisateur

Amazon Managed Service for Prometheus



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

Amazon Managed Service for Prometheus: Guide de l’utilisateur

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

Les marques et la présentation commerciale d’Amazon ne peuvent être utilisées en relation avec un produit ou un service qui n’est pas d’Amazon, d’une manière susceptible de créer une confusion parmi les clients, ou d’une manière qui dénigre ou discrédite Amazon. Toutes les autres marques commerciales qui ne sont pas la propriété d’Amazon appartiennent à leurs propriétaires respectifs, qui peuvent ou non être affiliés ou connectés à Amazon, ou sponsorisés par Amazon.

Table of Contents

Qu'est-ce qu'Amazon Managed Service for Prometheus ? 1
Régions prises en charge 1
Tarification 12
Support premium 13
Mise en route 14
Configuration AWS 14
Inscrivez-vous pour un Compte AWS 14
Création d'un espace de travail 15
Ingestion de métriques 16
Étape 1 : Ajouter de nouveaux référentiels de Charts Helm 17
Étape 2 : Créer un espace de noms Prometheus 17
Étape 3 : Configurer des rôles IAM pour les comptes de service 18
Étape 4 : Configurer le nouveau serveur et commencer à ingérer des métriques 18
Métriques des requêtes 19
Gérez les espaces de travail 22
Création d'un espace de travail 22
Configurez votre espace de travail 25
Modifier un alias d'espace de travail 28
Trouvez les détails de votre espace de travail 28
Suppression d'un espace de travail 30
Ingestion de métriques 32
AWS collectionneurs gérés 33
Intégrer Amazon EKS 34
Intégrer Amazon MSK 54
Intégrez Amazon OpenSearch Service 84
Prometheus-compatible indicateurs 97
Collecteurs de moniteurs 98
Collecteurs gérés par le client 104
Sécurisation de l'ingestion de vos métriques 104
Collecteurs ADOT 105
Collecteurs Prometheus 123
High-availability données 132
Interrogation de vos métriques 141
Aide-mémoire ProMQL 142

Sélecteurs de base 142
Sélecteurs vectoriels de plage 142
Opérateurs d'agrégation 143
Fonctions courantes 143
Opérateurs binaires 144
Exemples de requêtes pratiques 144
Sécurisez vos requêtes métriques 145
Utilisation AWS PrivateLink avec Amazon Managed Service pour Prometheus 105
Authentification et autorisation 105
Utiliser Amazon Managed Grafana 146
Connexion à Amazon Managed Grafana dans un VPC privé 146
Utilisez l'open source Grafana 147
Conditions préalables 147
Étape 1 : Configurer AWS SigV4 148
Étape 2 : Ajouter la source de données Prometheus dans Grafana 149
Étape 3 : (facultatif) Résolution des problèmes si la sauvegarde du & test ne fonctionne pas 151
Utiliser Grafana dans Amazon EKS 152
Configuration AWS SigV4 153
Configuration de rôles IAM pour les comptes de service 153
Mise à niveau du serveur Grafana à l'aide de Helm 155
Ajout de la source de données Prometheus dans Grafana 155
Utiliser des requêtes directes 156
Requête avec awscurl 157
Statistiques des requêtes 159
Détection des anomalies 163
Comment fonctionne la détection d'anomalies 163
Mise en route avec la détection des anomalies 164
PreviewAnomalyDetector 165
Formatage des paramètres de requête 165
Demande et réponse d'API 165
Règles d'enregistrement et d'alerte 169
Autorisations IAM nécessaires 170
Création d'un fichier de règles 171
Charger un fichier de règles 173
Modifier un fichier de règles 175

Résoudre les problèmes liés aux évaluations des règles 176
Valider l'état de déclenchement des alertes 177
Résoudre les notifications d'alerte manquantes 177
Vérifier l'état de santé des règles 178
Utiliser le décalage dans les requêtes pour gérer les délais d'ingestion 180
Problèmes courants et solutions correspondantes 180
Bonnes pratiques pour l'évaluation des règles 181
Dépannage des règles 182
Gestionnaire d'alertes 184
Autorisations IAM nécessaires 185
Création d'un fichier de configuration 186
Configuration d'un récepteur d'alerte 188
Amazon SNS 189
PagerDuty 199
Charger un fichier de configuration 205
Intégrez les alertes à Grafana 208
Conditions préalables 208
Configuration d'Amazon Managed Grafana 210
Résoudre les problèmes liés au gestionnaire d'alertes 211
Alertes actives (avertissement) 211
Avertissement concernant la taille du groupe d'agrégation d'alertes 212
La taille des alertes est trop grande (avertissement) 213
Avertissement de contenu vide 213
Avertissement `key/value` non valide 213
Avertissement de limite de message 214
Aucune erreur de stratégie basée sur les ressources 215
Avertissement de format non ASCII 215
Non autorisé à appeler KMS 216
Erreur de modèle 216
Surveillance des espaces de travail 218
CloudWatch métriques 218
Régler une CloudWatch alarme 236
CloudWatch Journaux 237
Configuration des CloudWatch journaux 237
Informations et contrôle des requêtes 239
Configuration de la journalisation des requêtes 240

Configuration des seuils de limitation des requêtes 242
Contenu du journal 243
Limitations 243
Compréhension et optimisation des coûts 245
Qu'est-ce qui contribue à mes coûts ? 245
Quel est le meilleur moyen de réduire mes coûts ? Comment réduire les coûts d'ingestion ? 245
Quel est le meilleur moyen de réduire mes coûts de requête ? 245
Si je réduis la période de conservation de mes métriques, cela contribuera-t-il à réduire ma facture totale ? 246
Comment puis-je réduire le coût de mes requêtes d'alerte ? 246
Puis-je consulter ma facture à tout moment ? 247
Quelles métriques puis-je utiliser pour surveiller mes coûts ? 247
Comment puis-je consulter mes coûts dans AWS Cost Explorer? 249
Comment calculer le nombre d'échantillons ingérés en un mois ? 250
Quelle granularité des données est disponible pour l'analyse des coûts historiques ? 252
Quelles sont les meilleures pratiques pour surveiller les coûts d'Amazon Managed Service for Prometheus ? 252
Pourquoi ma facture est-elle plus élevée en début de mois qu'en fin de mois ? 253
J'ai supprimé tous mes espaces de travail Amazon Managed Service for Prometheus, mais il semblerait que je sois toujours débité. Qu'est-ce qui pourrait se passer ? 254
Intégrations 255
Suivi des coûts Amazon EKS 255
AWS Accélérateur d'observabilité 256
Conditions préalables 256
Utilisation de l'exemple de métriques gérées (sans agent) 257
Alternative : collecteur autogéré OpenTelemetry 259
Affichage des tableaux de bord 260
AWS Contrôleurs pour Kubernetes 260
Conditions préalables 261
Déploiement d'un espace de travail 262
Configuration du cluster pour l'écriture à distance 266
Statistiques CloudWatch Amazon avec Firehose 268
Infrastructure 268
Création d'un CloudWatch stream Amazon 271
Nettoyage 272
Sécurité 273

Protection des données 274
Données collectées par Amazon Managed Service for Prometheus 275
Chiffrement au repos 276
Gestion de l'identité et des accès 289
Public ciblé 290
Authentification par des identités 290
Gestion de l'accès à l'aide de politiques 292
Utilisation d'Amazon Managed Service for Prometheus avec IAM 294
Exemples de politiques basées sur l'identité 300
Résolution des problèmes 303
Autorisations et politiques IAM 305
Autorisations Amazon Managed Service for Prometheus 305
Exemple de politiques IAM 305
Validation de la conformité 306
Résilience 307
Sécurité de l'infrastructure 307
Utilisation des rôles liés à un service 308
Rôle de récupération de métriques 308
CloudTrail journaux 310
Événements de gestion d'Amazon Managed Service for Prometheus dans CloudTrail 312
Exemples d'événements Amazon Managed Service pour Prometheus 313
Configuration de rôles IAM pour les comptes de service 317
Configuration de rôles de service pour l'ingestion de métriques à partir de clusters Amazon EKS 318
Configuration de rôles IAM de comptes de service pour l'interrogation des métriques 321
Points de terminaison de VPC d'Interface 324
Création d'un point de terminaison de VPC d'interface pour Amazon Managed Service for Prometheus 325
Résolution des problèmes 328
429 ou limite d'erreurs dépassées 328
Je vois des exemples en double. 330
Je vois des erreurs concernant les horodatages des échantillons 330
Je vois un message d'erreur lié à une limite. 330
La sortie de votre serveur Prometheus local dépasse la limite. 331
Certaines de mes données n'apparaissent pas 332
Erreur HTTP 422 : délai d'expiration de la requête 335

Identification 336
Identification des espaces de travail 337
Ajout d'une balise à un espace de travail 338
Visualisation des balises d'un espace de travail 340
Modification des balises d'un espace de travail 341
Suppression d'une balise d'un espace de travail 342
Identification des espaces de noms de groupes de règles 343
Ajouter une balise à un espace de noms de groupes de règles 344
Visualisation des balises d'un espace de noms de groupes de règles 346
Modification des balises d'un espace de noms de groupes de règles 347
Suppression d'une balise d'un espace de noms de groupes de règles 348
Quotas de service 351
Quotas de service 351
Quotas par défaut des séries actives 360
Dépasser le quota par défaut 361
Régulation de l'ingestion 361
Limites supplémentaires relatives aux données ingérées 363
Référence d'API 364
Amazon Managed Service pour Prometheus APIs 364
Utilisation d'Amazon Managed Service pour Prometheus avec un SDK AWS 365
Prometheus-compatible APIs 365
CreateAlertManagerAlerts 366
DeleteAlertManagerSilence 367
GetAlertManagerStatus 368
GetAlertManagerSilence 369
GetLabels 371
GetMetricMetadata 373
GetSeries 374
ListAlerts 376
ListAlertManagerAlerts 377
ListAlertManagerAlertGroups 379
ListAlertManagerReceivers 381
ListAlertManagerSilences 382
ListRules 384
PutAlertManagerSilences 386
QueryMetrics 388

RemoteWrite 390
Historique du document 392
..... cd

Qu'est-ce qu'Amazon Managed Service for Prometheus ?

Amazon Managed Service pour Prometheus est un service sans serveur, Prometheus-compatible service de surveillance des métriques de conteneurs qui facilite la surveillance sécurisée des environnements de conteneurs à grande échelle. Avec Amazon Managed Service for Prometheus, vous pouvez utiliser le même modèle de données et le même langage de requête open source Prometheus que vous utilisez aujourd'hui pour surveiller les performances de vos charges de travail conteneurisées et bénéficier d'une évolutivité, d'une disponibilité et d'une sécurité améliorées sans avoir à gérer l'infrastructure sous-jacente.

Amazon Managed Service for Prometheus adapte automatiquement l'ingestion, le stockage et l'interrogation des métriques opérationnelles à mesure que les charges de travail augmentent ou diminuent. Il s'intègre à AWS des services de sécurité pour permettre un accès rapide et sécurisé aux données.

Amazon Managed Service pour Prometheus est conçu pour être hautement disponible en utilisant plusieurs zones de disponibilité (Multi-AZ) déploiements. Les données ingérées dans un espace de travail sont répliquées dans trois zones de disponibilité de la même région.

Amazon Managed Service for Prometheus fonctionne avec des clusters de conteneurs qui s'exécutent sur Amazon Elastic Kubernetes Service et des environnements Kubernetes autogérés.

Avec Amazon Managed Service for Prometheus, vous utilisez le même modèle de données Prometheus open source et le même langage de requête ProMQL que vous utilisez avec Prometheus. Les équipes d'ingénierie peuvent utiliser ProMQL pour filtrer, agréger et générer des alarmes en fonction des métriques et obtenir rapidement une visibilité sur les performances sans aucune modification du code. Amazon Managed Service for Prometheus fournit des fonctionnalités de requête flexibles, sans coûts opérationnels ni complexité.

Les métriques ingérées dans un espace de travail sont stockées pendant 150 jours par défaut, puis sont automatiquement supprimées. Vous pouvez ajuster la période de conservation en configurant votre espace de travail jusqu'à un maximum de 1 095 jours (trois ans). Pour plus d'informations, voir Configurez votre espace de travail.

Régions prises en charge

Amazon Managed Service for Prometheus prend actuellement en charge les régions suivantes :

Nom de la région	Région	Point de terminaison	Protocole
USA Est (Ohio)	us-east-2	aps.us-east-2.amazonaws.com	HTTPS
		aps-workspaces.us-east-2.amazonaws.com	HTTPS
		aps-workspaces-fips.us-east-2.amazonaws.com	HTTPS
		aps-workspaces-fips.us-east-2.api.aws	HTTPS
		aps-workspaces.us-east-2.api.aws	HTTPS
		aps-fips.us-east-2.amazonaws.com	HTTPS
		aps.us-east-2.api.aws	HTTPS
		aps-fips.us-east-2.api.aws	HTTPS
USA Est (Virginie du Nord)	us-east-1	aps.us-east-1.amazonaws.com	HTTPS
		aps-workspaces.us-east-1.amazonaws.com	HTTPS
		aps-workspaces-fips.us-east-1.amazonaws.com	HTTPS
		aps-workspaces-fips.us-east-1.api.aws	HTTPS
		aps-workspaces.us-east-1.api.aws	HTTPS
		aps-fips.us-east-1.amazonaws.com	HTTPS
		aps.us-east-1.api.aws	HTTPS
		aps-fips.us-east-1.api.aws	HTTPS
USA Ouest (Californie du Nord)	us-west-1	aps.us-west-1.amazonaws.com	HTTPS
		aps-workspaces.us-west-1.amazonaws.com	HTTPS
			HTTPS

Nom de la région	Région	Point de terminaison	Protocole
		aps-workspaces-fips.us-west-1.amazonaws.com	HTTPS
		aps-workspaces-fips.us-west-1.api.aws	HTTPS
		aps-workspaces.us-west-1.api.aws	HTTPS
		aps-fips.us-west-1.amazonaws.com	HTTPS
		aps.us-west-1.api.aws	HTTPS
		aps-fips.us-west-1.api.aws	HTTPS
USA Ouest (Oregon)	us-west-2	aps.us-west-2.amazonaws.com	HTTPS
		aps-workspaces.us-west-2.amazonaws.com	HTTPS
		aps-workspaces-fips.us-west-2.amazonaws.com	HTTPS
		aps-workspaces-fips.us-west-2.api.aws	HTTPS
		aps-workspaces.us-west-2.api.aws	HTTPS
		aps-fips.us-west-2.amazonaws.com	HTTPS
		aps.us-west-2.api.aws	HTTPS
		aps-fips.us-west-2.api.aws	HTTPS
Afrique (Le Cap)	af-south-1	aps.af-south-1.amazonaws.com	HTTPS
		aps-workspaces.af-south-1.amazonaws.com	HTTPS
		aps-workspaces.af-south-1.api.aws	HTTPS
		aps.af-south-1.api.aws	HTTPS

Nom de la région	Région	Point de terminaison	Protocole
Asie-Pacifique (Hong Kong)	ap-east-1	aps.ap-east-1.amazonaws.com	HTTPS
		aps-workspaces.ap-east-1.amazonaws.com	HTTPS
		aps-workspaces.ap-east-1.api.aws	HTTPS
		aps.ap-east-1.api.aws	HTTPS
Asie-Pacifique (Hyderabad)	ap-south-2	aps.ap-south-2.amazonaws.com	HTTPS
		aps-workspaces.ap-south-2.amazonaws.com	HTTPS
		aps-workspaces.ap-south-2.api.aws	HTTPS
		aps.ap-south-2.api.aws	HTTPS
Asie-Pacifique (Jakarta)	ap-southeast-3	aps.ap-southeast-3.amazonaws.com	HTTPS
		aps-workspaces.ap-southeast-3.amazonaws.com	HTTPS
		aps-workspaces.ap-southeast-3.api.aws	HTTPS
		aps.ap-southeast-3.api.aws	HTTPS
Asie-Pacifique (Malaisie)	ap-southeast-5	aps.ap-southeast-5.amazonaws.com	HTTPS
		aps-workspaces.ap-southeast-5.amazonaws.com	HTTPS
		aps-workspaces.ap-southeast-5.api.aws	HTTPS
		aps.ap-southeast-5.api.aws	HTTPS

Nom de la région	Région	Point de terminaison	Protocole
Asie-Pacifique (Melbourne)	ap-southeast-4	aps.ap-southeast-4.amazonaws.com aps-workspaces.ap-southeast-4.amazonaws.com aps-workspaces.ap-southeast-4.api.aws aps.ap-southeast-4.api.aws	HTTPS HTTPS HTTPS HTTPS
Asie-Pacifique (Mumbai)	ap-south-1	aps.ap-south-1.amazonaws.com aps-workspaces.ap-south-1.amazonaws.com aps-workspaces.ap-south-1.api.aws aps.ap-south-1.api.aws	HTTPS HTTPS HTTPS HTTPS
Asie-Pacifique (Nouvelle Zélande)	ap-southeast-6	aps.ap-southeast-6.amazonaws.com aps-workspaces.ap-southeast-6.amazonaws.com aps-workspaces.ap-southeast-6.api.aws aps.ap-southeast-6.api.aws	HTTPS HTTPS HTTPS HTTPS
Asie-Pacifique (Osaka)	ap-northeast-3	aps.ap-northeast-3.amazonaws.com aps-workspaces.ap-northeast-3.amazonaws.com aps-workspaces.ap-northeast-3.api.aws aps.ap-northeast-3.api.aws	HTTPS HTTPS HTTPS HTTPS

Nom de la région	Région	Point de terminaison	Protocole
Asie-Pacifique (Séoul)	ap-northeast-2	aps.ap-northeast-2.amazonaws.com	HTTPS
		aps-workspaces.ap-northeast-2.amazonaws.com	HTTPS
		aps-workspaces.ap-northeast-2.api.aws	HTTPS
		aps.ap-northeast-2.api.aws	HTTPS
Asie-Pacifique (Singapour)	ap-southeast-1	aps.ap-southeast-1.amazonaws.com	HTTPS
		aps-workspaces.ap-southeast-1.amazonaws.com	HTTPS
		aps-workspaces.ap-southeast-1.api.aws	HTTPS
		aps.ap-southeast-1.api.aws	HTTPS
Asie-Pacifique (Sydney)	ap-southeast-2	aps.ap-southeast-2.amazonaws.com	HTTPS
		aps-workspaces.ap-southeast-2.amazonaws.com	HTTPS
		aps-workspaces.ap-southeast-2.api.aws	HTTPS
		aps.ap-southeast-2.api.aws	HTTPS
Asie-Pacifique (Taipei)	ap-east-2	aps.ap-east-2.amazonaws.com	HTTPS
		aps-workspaces.ap-east-2.amazonaws.com	HTTPS
		aps-workspaces.ap-east-2.api.aws	HTTPS
		aps.ap-east-2.api.aws	HTTPS

Nom de la région	Région	Point de terminaison	Protocole
Asie-Pacifique (Thaïlande)	ap-southeast-7	aps.ap-southeast-7.amazonaws.com aps-workspaces.ap-southeast-7.amazonaws.com aps-workspaces.ap-southeast-7.api.aws aps.ap-southeast-7.api.aws	HTTPS HTTPS HTTPS HTTPS
Asie-Pacifique (Tokyo)	ap-northeast-1	aps.ap-northeast-1.amazonaws.com aps-workspaces.ap-northeast-1.amazonaws.com aps-workspaces.ap-northeast-1.api.aws aps.ap-northeast-1.api.aws	HTTPS HTTPS HTTPS HTTPS
Canada (Centre)	ca-central-1	aps.ca-central-1.amazonaws.com aps-workspaces.ca-central-1.amazonaws.com aps-workspaces-fips.ca-central-1.amazonaws.com aps-workspaces-fips.ca-central-1.api.aws aps-workspaces.ca-central-1.api.aws aps-fips.ca-central-1.amazonaws.com aps.ca-central-1.api.aws aps-fips.ca-central-1.api.aws	HTTPS HTTPS HTTPS HTTPS HTTPS HTTPS HTTPS HTTPS

Nom de la région	Région	Point de terminaison	Protocole
Canada-Ouest (Calgary)	ca-west-1	aps.ca-west-1.amazonaws.com	HTTPS
		aps-workspaces.ca-west-1.amazonaws.com	HTTPS
		aps-workspaces-fips.ca-west-1.amazonaws.com	HTTPS
		aps-workspaces-fips.ca-west-1.api.aws	HTTPS
		aps-workspaces.ca-west-1.api.aws	HTTPS
		aps-fips.ca-west-1.amazonaws.com	HTTPS
		aps.ca-west-1.api.aws	HTTPS
		aps-fips.ca-west-1.api.aws	HTTPS
Europe (Francfort)	eu-central-1	aps.eu-central-1.amazonaws.com	HTTPS
		aps-workspaces.eu-central-1.amazonaws.com	HTTPS
		aps-workspaces.eu-central-1.api.aws	HTTPS
		aps.eu-central-1.api.aws	HTTPS
Europe (Irlande)	eu-west-1	aps.eu-west-1.amazonaws.com	HTTPS
		aps-workspaces.eu-west-1.amazonaws.com	HTTPS
		aps-workspaces.eu-west-1.api.aws	HTTPS
		aps.eu-west-1.api.aws	HTTPS

Nom de la région	Région	Point de terminaison	Protocole
Europe (Londres)	eu-west-2	aps.eu-west-2.amazonaws.com	HTTPS
		aps-workspaces.eu-west-2.amazonaws.com	HTTPS
		aps-workspaces.eu-west-2.api.aws	HTTPS
		aps.eu-west-2.api.aws	HTTPS
Europe (Milan)	eu-south-1	aps.eu-south-1.amazonaws.com	HTTPS
		aps-workspaces.eu-south-1.amazonaws.com	HTTPS
		aps-workspaces.eu-south-1.api.aws	HTTPS
		aps.eu-south-1.api.aws	HTTPS
Europe (Paris)	eu-west-3	aps.eu-west-3.amazonaws.com	HTTPS
		aps-workspaces.eu-west-3.amazonaws.com	HTTPS
		aps-workspaces.eu-west-3.api.aws	HTTPS
		aps.eu-west-3.api.aws	HTTPS
Europe (Espagne)	eu-south-2	aps.eu-south-2.amazonaws.com	HTTPS
		aps-workspaces.eu-south-2.amazonaws.com	HTTPS
		aps-workspaces.eu-south-2.api.aws	HTTPS
		aps.eu-south-2.api.aws	HTTPS
Europe (Stockholm)	eu-north-1	aps.eu-north-1.amazonaws.com	HTTPS
		aps-workspaces.eu-north-1.amazonaws.com	HTTPS
		aps-workspaces.eu-north-1.api.aws	HTTPS
		aps.eu-north-1.api.aws	HTTPS

Nom de la région	Région	Point de terminaison	Protocole
Europe (Zurich)	eu-central-2	aps.eu-central-2.amazonaws.com	HTTPS
		aps-workspaces.eu-central-2.amazonaws.com	HTTPS
		aps-workspaces.eu-central-2.api.aws	HTTPS
		aps.eu-central-2.api.aws	HTTPS
Israël (Tel Aviv)	il-central-1	aps.il-central-1.amazonaws.com	HTTPS
		aps-workspaces.il-central-1.amazonaws.com	HTTPS
		aps-workspaces.il-central-1.api.aws	HTTPS
		aps.il-central-1.api.aws	HTTPS
Mexique (Centre)	mx-central-1	aps.mx-central-1.amazonaws.com	HTTPS
		aps-workspaces.mx-central-1.amazonaws.com	HTTPS
		aps-workspaces.mx-central-1.api.aws	HTTPS
		aps.mx-central-1.api.aws	HTTPS
Moyen-Orient (Bahreïn)	me-south-1	aps.me-south-1.amazonaws.com	HTTPS
		aps-workspaces.me-south-1.amazonaws.com	HTTPS
		aps-workspaces.me-south-1.api.aws	HTTPS
		aps.me-south-1.api.aws	HTTPS
Moyen-Orient (EAU)	me-central-1	aps.me-central-1.amazonaws.com	HTTPS
		aps-workspaces.me-central-1.amazonaws.com	HTTPS
		aps-workspaces.me-central-1.api.aws	HTTPS
		aps.me-central-1.api.aws	HTTPS

Nom de la région	Région	Point de terminaison	Protocole
Amérique du Sud (São Paulo)	sa-east-1	aps.sa-east-1.amazonaws.com	HTTPS
		aps-workspaces.sa-east-1.amazonaws.com	HTTPS
		aps-workspaces.sa-east-1.api.aws	HTTPS
		aps.sa-east-1.api.aws	HTTPS
AWS GovCloud (US-East)	us-gov-east-1	aps.us-gov-east-1.amazonaws.com	HTTPS
		aps-workspaces.us-gov-east-1.amazonaws.com	HTTPS
		aps-workspaces-fips.us-gov-east-1.amazonaws.com	HTTPS
		aps-workspaces-fips.us-gov-east-1.api.aws	HTTPS
		aps-workspaces.us-gov-east-1.api.aws	HTTPS
		aps-fips.us-gov-east-1.amazonaws.com	HTTPS
		aps.us-gov-east-1.api.aws	HTTPS
		aps-fips.us-gov-east-1.api.aws	

Nom de la région	Région	Point de terminaison	Protocole
AWS GovCloud (US-West)	us-gov-west-1	aps.us-gov-west-1.amazonaws.com	HTTPS
		aps-workspaces.us-gov-west-1.amazonaws.com	HTTPS
		aps-workspaces-fips.us-gov-west-1.amazonaws.com	HTTPS
		aps-workspaces-fips.us-gov-west-1.api.aws	HTTPS
		aps-workspaces.us-gov-west-1.api.aws	HTTPS
		aps-fips.us-gov-west-1.amazonaws.com	HTTPS
		aps.us-gov-west-1.api.aws	HTTPS
		aps-fips.us-gov-west-1.api.aws	

Amazon Managed Service pour Prometheus inclut des points de terminaison du plan de contrôle (pour effectuer des tâches de gestion de l'espace de travail) et des points de terminaison du plan de données (pour fonctionner avec Prometheus-compatible données dans une instance d'espace de travail). Les points de terminaison du plan de contrôle commencent par `aps.*`, et les points de terminaison du plan de données commencent par `aps-workspaces.*`. Points de terminaison qui se terminent par `.amazonaws.com` prennent en charge IPv4 et les terminaux qui se terminent par `.api.aws` supporte à la fois IPv4 et IPv6.

Tarification

Vous devez payer des frais pour l'ingestion et le stockage des métriques. Les frais de stockage sont basés sur la taille compressée des échantillons de métriques et des métadonnées. Pour plus d'informations, consultez la section Amazon Managed Service for Prometheus Pricing.

Vous pouvez utiliser AWS Cost Explorer et AWS Rapports sur les coûts et l'utilisation pour suivre vos frais. Pour plus d'informations, voir Exploration de vos données à l'aide de Cost Explorer et Quels sont AWS Rapports sur les coûts et l'utilisation.

Support premium

Si vous vous abonnez à n'importe quel niveau du AWS plans de support premium, votre support premium s'applique à Amazon Managed Service pour Prometheus.

Commencez avec Amazon Managed Service pour Prometheus

Amazon Managed Service for Prometheus est un service sans serveur compatible avec Prometheus qui permet de surveiller les métriques des conteneurs. Il permet de surveiller facilement et en toute sécurité les environnements de conteneurs à grande échelle. Cette section décrit trois domaines clés de l'utilisation d'Amazon Managed Service pour Prometheus :

- Créez un espace de travail : créez un espace de travail Amazon Managed Service pour Prometheus afin de stocker et de surveiller vos statistiques.
- Ingestion des statistiques : votre espace de travail est vide jusqu'à ce que vous obteniez des statistiques dans votre espace de travail. Vous pouvez envoyer des métriques à Amazon Managed Service pour Prometheus, ou demander à Amazon Managed Service for Prometheus de supprimer automatiquement les métriques.
- Interrogez les métriques : une fois que vous avez des métriques sous forme de données dans votre espace de travail, vous êtes prêt à interroger les données pour explorer ou surveiller ces métriques.

Si vous êtes nouveau AWS, cette section inclut également des détails sur la configuration d'un Compte AWS.

Rubriques

- Configuration AWS
- Création d'un espace de travail Amazon Managed Service for Prometheus
- Intégration de métriques Prometheus dans l'espace de travail
- Interroger vos métriques Prometheus

Configuration AWS

Inscrivez-vous pour un Compte AWS

Pour commencer AWS, vous avez besoin d'un Compte AWS. Il existe deux manières de s'inscrire AWS. Si vous souhaitez contrôler totalement la configuration de votre compte et l'emplacement de

stockage de votre contenu, ou si vous avez des charges de travail réglementées telles que HIPAA ou FedRAMP, utilisez Inscrivez-vous pour AWS (avancé). Si vous êtes un nouvel utilisateur et que vous souhaitez utiliser un identifiant que vous possédez déjà pour vous connecter AWS et créez un AWS environnement avec des valeurs par défaut préconfigurées dans select Régions AWS, utiliser Inscrivez-vous pour AWS.

Création d'un espace de travail Amazon Managed Service for Prometheus

Un espace de travail est un espace logique dédié au stockage et à l'interrogation des métriques Prometheus. Un espace de travail prend en charge le contrôle d'accès à granularité fine pour autoriser sa gestion, notamment la mise à jour, la liste, la description, la suppression, ainsi que l'ingestion et l'interrogation de métriques. Vous pouvez avoir un ou plusieurs espaces de travail dans chaque région de votre compte.

Pour configurer un espace de travail, procédez comme suit.

> **Note**
>
> Pour plus d'informations sur la création d'un espace de travail et les options disponibles, consultezCréation d'un service géré Amazon pour l'espace de travail Prometheus.

Pour créer un espace de travail Amazon Managed Service for Prometheus

1. Ouvrez la console Amazon Managed Service for Prometheus à l'adresse. https://console.aws.amazon.com/prometheus/
2. Pour l'Alias d'espace de travail, entrez un alias pour le nouvel espace de travail.

   Les alias d'espace de travail sont des noms conviviaux qui vous aident à identifier vos espaces de travail. Ils ne doivent pas nécessairement être uniques. Deux espaces de travail peuvent avoir le même alias, mais tous les espaces de travail auront un espace de travail unique IDs, généré par Amazon Managed Service pour Prometheus.
3. (Facultatif) Pour ajouter des balises à l'espace de noms, choisissez Ajouter une nouvelle balise.

   Ensuite, pour Key (Clé), saisissez un nom de balise Vous pouvez ajouter une valeur en option pour la balise dans Value (Valeur).

Pour ajouter une autre étiquette, sélectionnez à nouveau Add new tag (Ajouter une nouvelle étiquette).

4. Choisissez Create workspace.

   La page de détails de l'espace de travail s'affiche. Cela affiche des informations telles que le statut, l'ARN, l'ID de l'espace de travail et le point de terminaison URLs de cet espace de travail pour les écritures à distance et les requêtes.

   Au départ, le statut est probablement CREATING. Attendez que le statut soit ACTIVE avant de passer à la configuration de votre ingestion de métriques.

   Prenez note des informations URLs affichées pour Endpoint (URL d'écriture à distance) et Endpoint (URL de requête). Vous en aurez besoin lorsque vous configurerez votre serveur Prometheus pour écrire à distance des métriques dans cet espace de travail et lorsque vous interrogerez ces métriques.

Intégration de métriques Prometheus dans l'espace de travail

L'un des moyens d'ingérer des métriques consiste à utiliser agent Prometheus autonome (une instance Prometheus exécutée en mode agent) pour extraire des métriques de votre cluster et les transmettre à Amazon Managed Service for Prometheus à des fins de stockage et de surveillance. Cette section explique comment configurer l'ingestion de métriques dans votre espace de travail Amazon Managed Service for Prometheus à partir d'Amazon EKS en configurant une nouvelle instance de l'agent Prometheus à l'aide de Helm.

Pour générer des métriques dans Amazon EKS, telles que Kubernetes ou des métriques au niveau des nœuds, vous pouvez utiliser les modules complémentaires de la communauté Amazon EKS. Pour plus d'informations, consultez la section Extensions communautaires disponibles dans le guide de l'utilisateur Amazon EKS.

Pour plus d'informations sur les autres méthodes d'ingestion de données dans Amazon Managed Service for Prometheus, notamment sur la manière de sécuriser les métriques et de créer des métriques haute disponibilité, consultez la section Ingérez les statistiques dans votre espace de travail Amazon Managed Service for Prometheus.

> **Note**
>
> Les métriques ingérées dans un espace de travail sont stockées pendant 150 jours par défaut, puis sont automatiquement supprimées. Vous pouvez ajuster la période de rétention en configurant votre espace de travail jusqu'à un maximum de 1 095 jours (trois ans). Pour plus d'informations, consultez Configurer votre espace de travail.

Les instructions de cette section vous permettent d'être rapidement opérationnel avec Amazon Managed Service for Prometheus. Cela suppose que vous avez déjà créé un espace de travail. Dans cette section, vous configurez un nouveau serveur Prometheus dans un cluster Amazon EKS, et le nouveau serveur utilise une configuration par défaut pour agir en tant qu'agent chargé d'envoyer des métriques à Amazon Managed Service for Prometheus. Voici les prérequis pour cette méthode :

- Vous devez disposer d'un cluster Amazon EKS à partir duquel le nouveau serveur Prometheus collectera les métriques.
- Un pilote Amazon EBS CSI doit être installé sur votre cluster Amazon EKS (requis par Helm).
- Vous devez utiliser Helm CLI 3.0 ou version ultérieure.
- Vous devez utiliser un ordinateur Linux ou macOS pour effectuer les étapes décrites dans les sections suivantes.

Étape 1 : Ajouter de nouveaux référentiels de Charts Helm

Pour ajouter de nouveaux référentiels de Charts de Helm, entrez les commandes suivantes. Pour plus d'informations sur l'utilisation de ces commandes, consultez la section Helm Repo.

```
helm repo add prometheus-community https://prometheus-community.github.io/helm-charts
helm repo add kube-state-metrics https://kubernetes.github.io/kube-state-metrics
helm repo update
```

Étape 2 : Créer un espace de noms Prometheus

Entrez la commande suivante pour créer un espace de noms Prometheus pour le serveur Prometheus et les autres composants de surveillance. *prometheus-agent-namespace*Remplacez-le par le nom que vous souhaitez pour cet espace de noms.

```
kubectl create namespace prometheus-agent-namespace
```

Étape 3 : Configurer des rôles IAM pour les comptes de service

Pour cette méthode d'ingestion, vous devez utiliser des rôles IAM pour les comptes de service du cluster Amazon EKS où l'agent Prometheus est exécuté.

Avec les rôles IAM pour les comptes de service, vous pouvez associer un rôle IAM à un compte de service Kubernetes. Ce compte de service peut ensuite fournir des AWS autorisations aux conteneurs de n'importe quel pod utilisant ce compte de service. Pour plus d'informations, consultez la section Rôles IAM pour les comptes de service.

Si vous n'avez pas encore configuré ces rôles, suivez les instructions de la section Configuration de rôles de service pour l'ingestion de métriques à partir de clusters Amazon EKS pour les configurer. Les instructions de cette section nécessitent l'utilisation de `eksctl`. Pour plus d'informations, consultez la section Démarrer avec Amazon Elastic Kubernetes Service – `eksctl`.

> **Note**
>
> Lorsque vous n'êtes pas sur EKS ou AWS que vous utilisez simplement une clé d'accès et une clé secrète pour accéder à Amazon Managed Service for Prometheus, vous ne pouvez pas utiliser `EKS-IAM-ROLE` le SigV4 basé.

Étape 4 : Configurer le nouveau serveur et commencer à ingérer des métriques

Pour installer le nouvel agent Prometheus et envoyer des métriques à votre espace de travail Amazon Managed Service for Prometheus, procédez comme suit.

Pour installer un nouvel agent Prometheus et envoyer des métriques à votre espace de travail Amazon Managed Service for Prometheus

1. À l'aide d'un éditeur de texte, créez un fichier nommé `my_prometheus_values_yaml` avec le contenu suivant.

   - *`IAM_PROXY_PROMETHEUS_ROLE_ARN`*Remplacez-le par l'ARN du amp-iamproxy-ingest-roleque vous avez créé dansConfiguration de rôles de service pour l'ingestion de métriques à partir de clusters Amazon EKS.
   - *`WORKSPACE_ID`*Remplacez-le par l'ID de votre espace de travail Amazon Managed Service for Prometheus.

- Remplacez *REGION* par la région de votre espace de travail Amazon Managed Service for Prometheus.

```
## The following is a set of default values for prometheus server helm chart which
 enable remoteWrite to AMP
## For the rest of prometheus helm chart values see: https://github.com/prometheus-
community/helm-charts/blob/main/charts/prometheus/values.yaml
##
serviceAccounts:
  server:
    name: amp-iamproxy-ingest-service-account
    annotations:
      eks.amazonaws.com/role-arn: ${IAM_PROXY_PROMETHEUS_ROLE_ARN}
server:
  remoteWrite:
    - url: https://aps-workspaces.${REGION}.amazonaws.com/workspaces/
${WORKSPACE_ID}/api/v1/remote_write
      sigv4:
        region: ${REGION}
      queue_config:
        max_samples_per_send: 1000
        max_shards: 200
        capacity: 2500
```

2. Saisissez la commande suivante pour créer le serveur Prometheus.

   - Remplacez *prometheus-chart-name* par le nom de votre version de Prometheus.
   - *prometheus-agent-namespace*Remplacez-le par le nom de votre espace de noms Prometheus.

```
helm install prometheus-chart-name prometheus-community/prometheus -n prometheus-
agent-namespace \
-f my_prometheus_values_yaml
```

Interroger vos métriques Prometheus

Maintenant que les métriques sont ingérées dans l'espace de travail, vous pouvez les interroger. Une méthode courante pour interroger vos métriques consiste à utiliser un service tel que Grafana. Dans

cette section, vous apprendrez à utiliser Amazon Managed Grafana pour interroger des métriques d'Amazon Managed Service pour Prometheus.

> **Note**
>
> Pour en savoir plus sur les autres moyens d'interroger votre Amazon Managed Service pour les métriques Prometheus, ou d'utiliser le service géré Amazon pour APIs Prometheus, consultez. Interroger vos métriques Prometheus

Cette section part du principe que vous avez déjà créé un espace de travail et que vous y ingérez des métriques.

Vous effectuez vos requêtes en utilisant le langage de requête standard de Prometheus, ProMQL. Pour plus d'informations sur ProMQL et sa syntaxe, consultez la section Querying Prometheus dans la documentation Prometheus.

Amazon Managed Grafana est un service entièrement géré pour Grafana open source qui simplifie la connexion à des éditeurs de logiciels indépendants indépendants et à des services permettant de visualiser AWS et d'analyser vos sources de données à grande échelle.

Amazon Managed Service for Prometheus prend en charge l'utilisation d'Amazon Managed Grafana pour interroger les métriques dans un espace de travail. Dans la console Amazon Managed Grafana, vous pouvez ajouter un espace de travail Amazon Managed Service for Prometheus en tant que source de données en découvrant vos comptes Amazon Managed Service for Prometheus existants. Amazon Managed Grafana gère la configuration des informations d'identification requises pour accéder à Amazon Managed Service for Prometheus. Pour obtenir des instructions détaillées sur la création d'une connexion à Amazon Managed Service for Prometheus à partir d'Amazon Managed Grafana, consultez les instructions du Guide de l'utilisateur Amazon Managed Grafana.

Vous pouvez également consulter vos alertes Amazon Managed Service for Prometheus dans Amazon Managed Grafana. Pour obtenir des instructions sur la configuration de l'intégration avec les alertes, consultez la section Intégrez des alertes à Amazon Managed Grafana ou à Grafana open source.

> **Note**
>
> Si vous avez configuré votre espace de travail Amazon Managed Grafana pour utiliser un VPC privé, vous devez connecter votre espace de travail Amazon Managed Service for

Prometheus au même VPC. Pour de plus amples informations, veuillez consulter Connexion à Amazon Managed Grafana dans un VPC privé.

Gérez les espaces de travail Amazon Managed Service pour Prometheus

Un espace de travail est un espace logique dédié au stockage et à l'interrogation des métriques Prometheus. Un espace de travail prend en charge le contrôle d'accès à granularité fine pour autoriser sa gestion, notamment la mise à jour, la liste, la description, la suppression, ainsi que l'ingestion et l'interrogation de métriques. Vous pouvez avoir un ou plusieurs espaces de travail dans chaque région de votre compte.

Suivez les procédures de cette section pour créer et gérer vos espaces de travail Amazon Managed Service for Prometheus.

Rubriques

- Création d'un service géré Amazon pour l'espace de travail Prometheus
- Configurez votre espace de travail
- Modifier un alias d'espace de travail
- Trouvez les détails de votre espace de travail Amazon Managed Service for Prometheus, y compris l'ARN
- Supprimer un espace de travail Amazon Managed Service pour Prometheus

Création d'un service géré Amazon pour l'espace de travail Prometheus

Pour créer un espace de travail Amazon Managed Service for Prometheus, procédez comme suit. Vous pouvez choisir d'utiliser la console AWS CLI ou Amazon Managed Service for Prometheus.

> **Note**
>
> Si vous utilisez un cluster Amazon EKS, vous pouvez également créer un nouvel espace de travail à l'aide de AWS Controllers for Kubernetes.

Pour créer un espace de travail à l'aide du AWS CLI

1. Saisissez la commande suivante pour créer l'espace de travail. Cet exemple crée un espace de travail nommé `my-first-workspace`, mais vous pouvez utiliser un autre alias (ou aucun alias) si vous le souhaitez. Les alias d'espace de travail sont des noms conviviaux qui vous aident à identifier vos espaces de travail. Ils ne doivent pas nécessairement être uniques. Deux espaces de travail peuvent avoir le même alias, mais tous les espaces de travail ont des ID d'espace de travail uniques, générés par Amazon Managed Service for Prometheus.

   (Facultatif) Pour utiliser votre propre clé KMS afin de chiffrer les données stockées dans votre espace de travail, vous pouvez inclure le `kmsKeyArn` paramètre dans la AWS KMS clé à utiliser. Bien qu'Amazon Managed Service for Prometheus ne vous facture pas l'utilisation de clés gérées par le client, des coûts peuvent être associés aux clés provenant de. AWS Key Management Service Pour plus d'informations sur le chiffrement des données dans l'espace de travail par Amazon Managed Service for Prometheus ou sur la création, la gestion et l'utilisation de votre propre clé gérée par le client, consultez Chiffrement au repos.

   Les paramètres entre crochets (`[]`) étant facultatifs, n'en incluez pas dans votre commande.

   ```
   aws amp create-workspace [--alias my-first-workspace] [--kmsKeyArn arn:aws:aps:us-west-2:111122223333:workspace/ws-sample-1234-abcd-56ef-7890abcd12ef]  [--tags Status=Secret,Team=My-Team]
   ```

   Cette commande renvoie les données suivantes :

   - `workspaceId` est l'ID unique de cet espace de travail. Notez cet ID.
   - `arn` est l'ARN de cet espace de travail.
   - `status` est le statut actuel de l'espace de travail. Immédiatement après la création de l'espace de travail, ce sera probablement `CREATING`.
   - `kmsKeyArn` est la clé gérée par le client utilisée pour chiffrer les données de l'espace de travail, si elle est fournie.

   > **Note**
   >
   > Les espaces de travail créés avec les clés gérées par le client ne peuvent pas utiliser les collecteurs gérés AWS pour l'ingestion.
   > Choisissez d'utiliser avec soin les clés gérées par le client ou les clés AWS détenues par le client. Les espaces de travail créés avec des clés gérées par le client ne

peuvent pas être convertis ultérieurement pour utiliser des clés AWS détenues (et vice versa).

- `tags` répertorie les balises de l'espace de travail, le cas échéant.

2. Si votre commande `create-workspace` renvoie le statut `CREATING`, vous pouvez alors entrer la commande suivante pour déterminer à quel moment l'espace de travail sera prêt. Remplacez *`my-workspace-id`* par la valeur renvoyée par la `create-workspace` commande`workspaceId`.

```
aws amp describe-workspace --workspace-id my-workspace-id
```

Lorsque la commande `describe-workspace` renvoie `ACTIVE` pour `status`, l'espace de travail est prêt à être utilisé.

Pour créer un espace de travail à l'aide de la console Amazon Managed Service for Prometheus

1. Ouvrez la console Amazon Managed Service for Prometheus à l'adresse. https://console.aws.amazon.com/prometheus/
2. Choisissez Créer.
3. Pour l'Alias d'espace de travail, entrez un alias pour le nouvel espace de travail.

   Les alias d'espace de travail sont des noms conviviaux qui vous aident à identifier vos espaces de travail. Ils ne doivent pas nécessairement être uniques. Deux espaces de travail peuvent avoir le même alias, mais tous les espaces de travail ont des ID d'espace de travail uniques, générés par Amazon Managed Service for Prometheus.
4. (Facultatif) Pour utiliser votre propre clé KMS pour chiffrer les données stockées dans votre espace de travail, vous pouvez sélectionner Personnaliser les paramètres de chiffrement et choisir la AWS KMS clé à utiliser (ou en créer une nouvelle). Vous pouvez choisir une clé de votre compte dans la liste déroulante ou saisir l'ARN d'une clé à laquelle vous avez accès. Bien qu'Amazon Managed Service for Prometheus ne vous facture pas l'utilisation de clés gérées par le client, des coûts peuvent être associés aux clés provenant de. AWS Key Management Service

   Pour plus d'informations sur le chiffrement des données dans l'espace de travail par Amazon Managed Service for Prometheus ou sur la création, la gestion et l'utilisation de votre propre clé gérée par le client, consultez Chiffrement au repos.

> **Note**
>
> Les espaces de travail créés avec les clés gérées par le client ne peuvent pas utiliser les collecteurs gérés AWS pour l'ingestion.
> Choisissez d'utiliser avec soin les clés gérées par le client ou les clés AWS détenues par le client. Les espaces de travail créés avec des clés gérées par le client ne peuvent pas être convertis ultérieurement pour utiliser des clés AWS détenues (et vice versa).

5. (Facultatif) Pour ajouter une ou plusieurs balises à l'espace de travail, choisissez Ajouter une nouvelle balise. Ensuite, dans Clé, saisissez un nom de balise. Vous pouvez ajouter une valeur en option pour la balise dans Value (Valeur).

   Pour ajouter une autre étiquette, sélectionnez à nouveau Add new tag (Ajouter une nouvelle étiquette).

6. Choisissez Create workspace.

   La page de détails de l'espace de travail s'affiche. Elle contient des informations telles que le statut, l'ARN, l'ID d'espace de travail et les URL des points de terminaison de cet espace de travail pour les écritures et requêtes à distance.

   Le statut renvoie CREATING jusqu'à ce que l'espace de travail soit prêt. Attendez que le statut soit ACTIVE avant de passer à la configuration de votre ingestion de métriques.

   Notez les URL affichées pour Endpoint - remote write URL et Endpoint - query URL. Vous en aurez besoin lorsque vous configurerez votre serveur Prometheus pour écrire à distance des métriques dans cet espace de travail et lorsque vous interrogerez ces métriques.

Pour plus d'informations sur l'ingestion de métriques dans l'espace de travail, consultez la section Intégration de métriques Prometheus dans l'espace de travail.

Configurez votre espace de travail

Vous pouvez configurer votre espace de travail comme suit :

- Définissez des jeux d'étiquettes et définissez des limites pour les séries chronologiques actives qui correspondent à vos jeux d'étiquettes définis. Un jeu d'étiquettes est un ensemble d'une ou

de plusieurs étiquettes, qui sont des name/value paires qui aident à donner du contexte aux statistiques des séries chronologiques.

En définissant des ensembles d'étiquettes et en fixant des limites de séries chronologiques actives, vous pouvez limiter les pics chez un locataire ou une source afin qu'ils n'affectent que ce locataire ou cette source. Par exemple, si vous définissez une limite de 1 000 000 de séries chronologiques actives sur le jeu d'étiquette`team=A env=prod`, si le nombre de séries chronologiques ingérées correspondant à ce jeu d'étiquettes dépasse la limite, seules les séries chronologiques correspondant au jeu d'étiquettes sont limitées. De cette façon, les autres locataires ou sources de mesures ne sont pas affectés.

Pour plus d'informations sur les étiquettes dans Prometheus, consultez la section Modèle de données.

- Définissez une période de conservation pour définir le nombre de jours pendant lesquels les données doivent être conservées dans l'espace de travail.
- Définissez une période de rupture de stock pour permettre à l'espace de travail d'accepter les échantillons qui arrivent hors ordre chronologique. Tous les espaces de travail ont une fenêtre de temps hors service par défaut de 60 secondes. La définition de cette valeur sur 0 désactive la fonctionnalité et les échantillons en rupture de commande sont supprimés.
- Définissez un décalage des requêtes de règles pour introduire un délai global avant l'exécution des requêtes d'évaluation des règles. Cela compense les retards d'ingestion ou les échantillons hors service, garantissant ainsi que les règles sont évaluées par rapport à un ensemble de données plus complet. Un décalage de requête configuré au niveau du groupe de règles est prioritaire par rapport à ce paramètre au niveau de l'espace de travail. Pour de plus amples informations, veuillez consulter Utiliser le décalage dans les requêtes pour gérer les délais d'ingestion.

Pour configurer votre espace de travail

1. Ouvrez la console Amazon Managed Service for Prometheus à l'adresse. https://console.aws.amazon.com/prometheus/
2. Dans le coin supérieur gauche de la page, cliquez sur l'icône du menu, puis sur Tous les espaces de travail.
3. Choisissez l'ID d'espace de travail.
4. Choisissez l'onglet Configurations de l'espace de travail.

5. Pour définir la période de conservation de l'espace de travail, choisissez Modifier dans la section Période de conservation. Spécifiez ensuite la nouvelle période de conservation en jours. Le maximum est de 1 095 jours (trois ans).
6. Pour définir la période de rupture de commande, choisissez Modifier dans la section Fenêtre de rupture de commande. Spécifiez ensuite la fenêtre temporelle en secondes. Tous les espaces de travail ont une fenêtre de temps hors service par défaut de 60 secondes. La valeur minimale est de 0 seconde ; lorsqu'elle est définie sur 0, les échantillons en rupture de commande sont éliminés. La valeur maximale est de 600 secondes. Pour demander un maximum plus élevé, consultezService Quotas d'Amazon Managed Service for Prometheus.
7. Pour définir le décalage de requête de règle, choisissez Modifier dans la section Décalage de requête de règle. Spécifiez ensuite le décalage en secondes. La valeur par défaut pour les nouveaux espaces de travail est de 60 secondes. La valeur minimale est de 0 seconde et la valeur maximale est de 86 400 secondes (24 heures).

   > **Note**
   >
   > Les espaces de travail existants ont un décalage de requête par défaut de 0 seconde, que vous pouvez mettre à jour via la console.
   > Si vous avez configuré une fenêtre temporelle hors ordre, pensez à définir le décalage de la requête de règle sur une valeur supérieure ou égale à la fenêtre temporelle hors service. Cela garantit que les échantillons hors service sont inclus dans les évaluations des règles. Le décalage d'une requête de règle retarde les évaluations des règles et les alertes qui en résultent de la durée configurée.

8. Pour ajouter ou modifier des ensembles d'étiquettes et leurs limites de séries actives, choisissez Modifier dans la section Ensembles d'étiquettes. Ensuite, procédez comme suit :

   a. (Facultatif) Entrez une valeur dans Limite de compartiment par défaut pour définir le nombre maximum de séries chronologiques actives pouvant être ingérées dans l'espace de travail, en comptant uniquement les séries chronologiques qui ne correspondent à aucun ensemble d'étiquettes défini.

   b. Pour définir un jeu d'étiquettes, entrez une limite de série chronologique active pour le nouveau jeu d'étiquettes sous Limite de série active.

      Entrez ensuite une étiquette et une valeur pour une étiquette qui sera utilisée dans le jeu d'étiquettes, puis choisissez Ajouter une étiquette.

c. (Facultatif) Pour définir un autre jeu d'étiquettes, choisissez Ajouter un autre jeu d'étiquettes et répétez les étapes précédentes.

9. Lorsque vous avez terminé, choisissez Save changes (Enregistrer les modifications).

Modifier un alias d'espace de travail

Vous pouvez modifier un espace de travail pour changer son alias. Pour modifier l'alias de l'espace de travail à l'aide de l' AWS CLI, saisissez la commande suivante.

```
aws amp update-workspace-alias --workspace-id my-workspace-id --alias "new-alias"
```

Pour modifier un espace de travail à l'aide de la console Amazon Managed Service for Prometheus

1. Ouvrez la console Amazon Managed Service for Prometheus à l'adresse. https://console.aws.amazon.com/prometheus/
2. Dans le coin supérieur gauche de la page, cliquez sur l'icône du menu, puis sur Tous les espaces de travail.
3. Choisissez l'ID de l'espace de travail à modifier, puis sélectionnez Modifier.
4. Entrez un nouvel alias pour l'espace de travail, puis sélectionnez Enregistrer.

Trouvez les détails de votre espace de travail Amazon Managed Service for Prometheus, y compris l'ARN

Vous pouvez trouver les détails de votre espace de travail Amazon Managed Service for Prometheus en utilisant la console ou AWS le. AWS CLI

Console

Pour trouver les détails de votre espace de travail à l'aide de la console Amazon Managed Service pour Prometheus

1. Ouvrez la console Amazon Managed Service for Prometheus à l'adresse. https://console.aws.amazon.com/prometheus/
2. Dans le coin supérieur gauche de la page, cliquez sur l'icône du menu, puis sur Tous les espaces de travail.

3. Choisissez l'ID d'espace de travail. Cela affichera des informations sur votre espace de travail, notamment :

   - État actuel : le statut de votre espace de travail, par exemple Actif, est affiché sous État.
   - ARN — L'ARN de l'espace de travail est affiché sous ARN.
   - ID — L'identifiant de l'espace de travail est affiché sous ID de l'espace de travail.
   - URL : la console affiche plusieurs URL pour l'espace de travail, y compris les URL permettant d'écrire ou d'interroger des données depuis l'espace de travail.

   > **Note**
   >
   > Par défaut, les URL indiquées sont les URL IPv4. Vous pouvez également utiliser des URL à double pile (IPv4 et IPv6 compatibles). Ce sont les mêmes, mais ils se trouvent dans le domaine `api.aws` plutôt que dans le domaine par défaut`amazonaws.com`. Par exemple, si vous deviez voir ce qui suit (une URL IPv4) :
   >
   > ```
   > https://aps-workspaces.us-east-1.amazonaws.com/workspaces/ws-abcd1234-ef56-7890-ab12-example/api/v1/remote_write
   > ```
   >
   > Vous pouvez créer une URL dualstack (y compris le support pour IPv6) comme suit :
   >
   > ```
   > https://aps-workspaces.us-east-1.api.aws/workspaces/ws-abcd1234-ef56-7890-ab12-example/api/v1/remote_write
   > ```

   Sous cette section se trouvent des onglets contenant des informations sur les règles, le gestionnaire d'alertes, les journaux, la configuration et les balises.

AWS CLI

Pour trouver les détails de votre espace de travail à l'aide du AWS CLI

La commande suivante renvoie les détails de l'espace de travail. Vous devez le *`my-workspace-id`* remplacer par l'ID de l'espace de travail pour lequel vous souhaitez obtenir des informations.

```
aws amp describe-workspace --workspace-id my-workspace-id
```

Cela renvoie des informations sur votre espace de travail, notamment :

- État actuel — Le statut de votre espace de travail, par exemple`ACTIVE`, est renvoyé dans la `statusCode` propriété.
- ARN — L'ARN de l'espace de travail est renvoyé dans la `arn` propriété.
- URL : AWS CLI renvoie l'URL de base de l'espace de travail de la `prometheusEndpoint` propriété.

> **Note**
>
> Par défaut, l'URL renvoyée est l'URL IPv4. Vous pouvez également utiliser une URL à double pile (IPv4 et IPv6 pris en charge) dans le domaine `api.aws` plutôt que l'URL par défaut. `amazonaws.com` Par exemple, si vous deviez voir ce qui suit (une URL IPv4) :
>
> ```
> https://aps-workspaces.us-east-1.amazonaws.com/workspaces/ws-abcd1234-ef56-7890-ab12-example/
> ```
>
> Vous pouvez créer une URL dualstack (y compris le support pour IPv6) comme suit :
>
> ```
> https://aps-workspaces.us-east-1.api.aws/workspaces/ws-abcd1234-ef56-7890-ab12-example/
> ```
>
> Vous pouvez également créer les URL d'écriture et de requête à distance pour l'espace de travail, en ajoutant `/api/v1/remote_write` ou`/api/v1/query`, respectivement.

Supprimer un espace de travail Amazon Managed Service pour Prometheus

La suppression d'un espace de travail entraîne la suppression des données qui y ont été ingérées.

> **Note**
>
> La suppression d'un espace de travail Amazon Managed Service for Prometheus ne supprime pas automatiquement les collecteurs gérés qui collectent AWS des métriques et

les envoient vers l'espace de travail. Pour de plus amples informations, veuillez consulter Recherche et suppression des scrapers.

Pour supprimer un espace de travail à l'aide du AWS CLI

Utilisez la commande suivante :

```
aws amp delete-workspace --workspace-id my-workspace-id
```

Pour supprimer un espace de travail à l'aide de la console Amazon Managed Service for Prometheus

1. Ouvrez la console Amazon Managed Service for Prometheus à l'adresse. https://console.aws.amazon.com/prometheus/
2. Dans le coin supérieur gauche de la page, cliquez sur l'icône du menu, puis sur Tous les espaces de travail.
3. Choisissez l'ID de l'espace de travail à supprimer, puis sélectionnez Supprimer.
4. Dans la zone de confirmation, saisissez **`delete`**, puis sélectionnez Supprimer.

Ingérez les statistiques dans votre espace de travail Amazon Managed Service for Prometheus

Les métriques doivent être ingérées dans votre espace de travail Amazon Managed Service for Prometheus pour que vous puissiez les interroger ou émettre des alertes à leur sujet. Cette section explique comment configurer l'ingestion de métriques dans votre espace de travail.

> **Note**
>
> Les métriques ingérées dans un espace de travail sont stockées pendant 150 jours par défaut, puis sont automatiquement supprimées. Vous pouvez ajuster la période de rétention en configurant votre espace de travail jusqu'à un maximum de 1 095 jours (trois ans). Pour plus d'informations, voir Configuration de votre espace de travail.

Il existe deux méthodes pour ingérer des métriques dans votre espace de travail Amazon Managed Service for Prometheus.

- À l'aide d'un collecteur AWS géré, Amazon Managed Service for Prometheus fournit un scraper entièrement géré et sans agent pour extraire automatiquement les métriques de vos clusters Amazon Elastic Kubernetes Service (Amazon EKS). La collecte extrait automatiquement les métriques des points de terminaison compatibles avec Prometheus.
- Utilisation d'un collecteur géré par le client – Vous disposez de nombreuses options pour gérer votre propre collecteur. Deux des collecteurs les plus courants sont l'installation de votre propre instance de Prometheus, l'exécution en mode agent ou AWS l'utilisation de Distro pour. OpenTelemetry Ils sont tous deux décrits en détail dans les sections suivantes.

  Les collecteurs envoient des métriques à Amazon Managed Service for Prometheus à l'aide de la fonctionnalité d'écriture à distance de Prometheus. Vous pouvez envoyer des métriques directement à Amazon Managed Service for Prometheus en utilisant l'écriture à distance Prometheus dans votre propre application. Pour plus de détails sur l'utilisation directe de l'écriture à distance et des configurations d'écriture à distance, consultez la section remote_write dans la documentation de Prometheus.

Rubriques

- Ingérez des métriques avec AWS collectionneurs gérés

- Collecteurs gérés par le client

Ingérez des métriques avec AWS collectionneurs gérés

Une des utilisations courantes d'Amazon Managed Service for Prometheus consiste à surveiller les clusters Kubernetes gérés par Amazon Elastic Kubernetes Service (Amazon EKS). Les clusters Kubernetes et de nombreuses applications exécutées dans Amazon EKS exportent automatiquement leurs métriques pour Prometheus-compatible grattoirs pour y accéder.

> **Note**
>
> Amazon EKS expose les métriques des serveurs d'API, `kube-controller-manager` des indicateurs, et `kube-scheduler` métriques dans un cluster. De nombreuses autres technologies et applications exécutées dans des environnements Kubernetes fournissent Prometheus-compatible métriques. Pour obtenir la liste des exportateurs les plus connus, consultez la section Exportateurs et intégrations de la Documentation de Prometheus.

Amazon Managed Service pour Prometheus fournit un scraper entièrement géré, sans agent, ou collectionneur, qui découvre et extrait automatiquement Prometheus-compatible métriques. Vous n'avez pas besoin de gérer, d'installer, de corriger ou de maintenir des agents ou des scrapers. Le collecteur Amazon Managed Service for Prometheus fournit une collecte de métriques fiable, stable, hautement disponible et automatiquement à l'échelle pour votre cluster Amazon EKS. Les collecteurs gérés Amazon Managed Service pour Prometheus fonctionnent avec les clusters Amazon EKS, notamment EC2 et Fargate.

Un collecteur Amazon Managed Service for Prometheus crée une Interface réseau Elastic (ENI) par sous-réseau spécifié lors de la création du scraper. Le collecteur extrait les métriques via ces ENI et utilise `remote_write` pour transférer les données vers votre espace de travail Amazon Managed Service for Prometheus à l'aide du point de terminaison d'un VPC. Les données scrapées ne sont jamais transmises sur l'Internet public.

Les rubriques suivantes fournissent des informations supplémentaires sur l'utilisation d'un collecteur Amazon Managed Service for Prometheus dans votre cluster Amazon EKS, ainsi que sur les métriques collectées.

Rubriques

- Configurer des collecteurs gérés pour Amazon EKS

- Configurer des collecteurs Prometheus gérés pour Amazon MSK
- Configurer des collecteurs Prometheus gérés pour Amazon OpenSearch Service
- Quels sont Prometheus-compatible des indicateurs ?
- Surveillez les collecteurs à l'aide de journaux vendus

Configurer des collecteurs gérés pour Amazon EKS

Pour utiliser un collecteur Amazon Managed Service for Prometheus, vous devez créer un scraper qui découvre et extrait les métriques de votre cluster Amazon EKS. Vous pouvez également créer un scraper qui s'intègre à Amazon Managed Streaming pour Apache, Kafka ou à Amazon. OpenSearch Service. Pour plus d'informations, voir Intégrez Amazon MSK et Intégrez Amazon OpenSearch Service.

- Vous pouvez créer un scraper dans le cadre de la création de votre cluster Amazon EKS. Pour plus d'informations sur la création d'un cluster Amazon EKS, notamment la création d'un scraper, consultez la section Création d'un cluster Amazon EKS dans le Guide de l'utilisateur Amazon EKS.
- Vous pouvez créer votre propre scraper par programmation à l'aide du AWS API ou en utilisant AWS CLI.

Un collecteur Amazon Managed Service for Prometheus extrait les métriques qui sont Prometheus-compatible. Pour plus d'informations sur les métriques compatibles avec Prometheus, consultez la section Quels sont Prometheus-compatible des indicateurs ?. Les clusters Amazon EKS exposent les métriques du serveur d'API. Clusters Amazon EKS qui sont des versions de Kubernetes `1.28` ou au-dessus, exposent également les métriques pour le `kube-scheduler` et `kube-controller-manager`. Pour en savoir plus, consultez Récupérer les métriques brutes du plan de contrôle au format Prometheus dans le Guide de l'utilisateur Amazon EKS.

> **Note**
>
> L'extraction de métriques d'un cluster peut entraîner des frais d'utilisation du réseau. L'un des moyens d'optimiser ces coûts est de configurer votre `/metrics` endpoint pour compresser les métriques fournies (par exemple, avec gzip), réduisant ainsi les données qui doivent être déplacées sur le réseau. La procédure à suivre dépend de l'application ou de la bibliothèque qui fournit les métriques. Certaines bibliothèques sont compressées par défaut.

Les rubriques suivantes décrivent comment créer, gérer et configurer des scrapers.

Rubriques

- Créer un scraper
- Configuration de votre cluster Amazon EKS
- Recherche et suppression des scrapers
- Configuration du scraper
- Résolution des erreurs de configuration du scraper
- Limitations du scraper

Créer un scraper

Un collecteur Amazon Managed Service for Prometheus consiste en un scraper qui reconnaît et collecte des métriques d'un cluster Amazon EKS. Amazon Managed Service for Prometheus gère le scraper pour vous, vous offrant ainsi l'évolutivité, la sécurité et la fiabilité dont vous avez besoin, sans avoir à gérer vous-même les instances, les agents ou les scrapers.

Il existe trois manières de créer un grattoir :

- Un grattoir est automatiquement créé pour vous lorsque vous créer un cluster Amazon EKS via la console Amazon EKS et choisissez d'activer les métriques Prometheus.
- Vous pouvez créer un scraper depuis la console Amazon EKS pour un cluster existant. Ouvrez le cluster dans Console Amazon EKS, puis, sur le Observabilité onglet, choisissez Ajouter un grattoir.

  Pour plus de détails sur les paramètres disponibles, voir Activer les métriques Prometheus dans le Guide de l'utilisateur d'Amazon EKS.
- Vous pouvez créer un grattoir à l'aide du AWS API ou AWS CLI.

  Ces options sont décrites dans la procédure suivante.

Il existe quelques prérequis pour créer votre propre scraper :

- Vous devez avoir créé un cluster Amazon EKS.
- Le contrôle d'accès aux points de terminaison du cluster de votre propre cluster Amazon EKS doit être configuré pour inclure l'accès privé. Il peut inclure l'accès privé et l'accès public, mais doit inclure l'accès privé.

- Le VPC Amazon dans lequel se trouve le cluster Amazon EKS doit avoir DNS activé.

> **Note**
>
> Le cluster sera associé au scraper par son nom de ressource Amazon (ARN). Si vous supprimez un cluster, puis que vous en créez un nouveau avec le même nom, l'ARN sera réutilisé pour le nouveau cluster. Pour cette raison, le scraper tentera de collecter des métriques pour le nouveau cluster. Vous supprimer les grattoirs séparément de la suppression du cluster.

AWS API

Pour créer un scraper à l'aide de l'API AWS

Utilisez le `CreateScraper` Fonctionnement de l'API pour créer un scraper avec AWS API. L'exemple suivant crée un scraper dans la région `us-west-2`. Vous devez remplacer les informations relatives à Compte AWS, l'espace de travail, la sécurité et au cluster Amazon EKS par vos propres identifiants, et fournir la configuration à utiliser pour votre scraper.

> **Note**
>
> Le groupe de sécurité et les sous-réseaux doivent être définis sur le groupe de sécurité et les sous-réseaux du cluster auquel vous vous connectez.
> Vous devez inclure au moins deux sous-réseaux dans au moins deux zones de disponibilité.

La `scrapeConfiguration` est un fichier YAML de configuration Prometheus codé en base64. Vous pouvez télécharger une configuration générale à l'aide de l'opération d'API `GetDefaultScraperConfiguration`. Pour plus d'informations sur le format du `scrapeConfiguration`, voir Configuration du scraper.

```
POST /scrapers HTTP/1.1
Content-Length: 415
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: aws-cli/1.18.147 Python/2.7.18 Linux/5.4.58-37.125.amzn2int.x86_64
 botocore/1.18.6
```

```
{
    "alias": "myScraper",
    "destination":  {
        "ampConfiguration": {
            "workspaceArn": "arn:aws:aps:us-west-2:account-id:workspace/
ws-workspace-id"
        }
    },
    "source": {
        "eksConfiguration": {
            "clusterArn": "arn:aws:eks:us-west-2:account-id:cluster/cluster-name",
            "securityGroupIds": ["sg-security-group-id"],
            "subnetIds": ["subnet-subnet-id-1", "subnet-subnet-id-2"]
        }
    },
    "scrapeConfiguration": {
        "configurationBlob": <base64-encoded-blob>
    }
}
```

AWS CLI

Pour créer un scraper à l'aide de l' AWS CLI

Utilisez le `create-scraper` commande pour créer un grattoir avec le AWS CLI. L'exemple suivant crée un scraper dans la région `us-west-2`. Vous devez remplacer les informations relatives à Compte AWS, l'espace de travail, la sécurité et au cluster Amazon EKS par vos propres identifiants, et fournir la configuration à utiliser pour votre scraper.

> **Note**
>
> Le groupe de sécurité et les sous-réseaux doivent être définis sur le groupe de sécurité et les sous-réseaux du cluster auquel vous vous connectez.
> Vous devez inclure au moins deux sous-réseaux dans au moins deux zones de disponibilité.

La `scrape-configuration` est un fichier YAML de configuration Prometheus codé en base64. Vous pouvez télécharger une configuration générale à l'aide du `get-default-`

`scraper-configuration` commande. Pour plus d'informations sur le format du `scrape-configuration`, voir Configuration du scraper.

```
aws amp create-scraper \
  --source eksConfiguration="{clusterArn='arn:aws:eks:us-west-2:account-id:cluster/cluster-name', securityGroupIds=['sg-security-group-id'],subnetIds=['subnet-subnet-id-1', 'subnet-subnet-id-2']}" \
  --scrape-configuration configurationBlob=<base64-encoded-blob> \
  --destination ampConfiguration="{workspaceArn='arn:aws:aps:us-west-2:account-id:workspace/ws-workspace-id'}"
```

Vous trouverez ci-dessous la liste complète des opérations de scraper que vous pouvez utiliser avec l'API AWS :

- Créez un scraper avec l'opération d'API CreateScraper.
- Répertoriez vos scrapers existants avec l'opération d'API ListScrapers.
- Mettez à jour l'alias, la configuration ou la destination d'un scraper avec UpdateScraper Fonctionnement de l'API.
- Supprimez un scraper avec l'opération d'API DeleteScraper.
- Obtenez plus de détails sur un scraper avec l'opération d'API DescribeScraper.
- Obtenez une configuration générale pour les scrapers avec l'opération d'API GetDefaultScraperConfiguration.

> **Note**
>
> Le cluster Amazon EKS que vous collectez doit être configuré pour autoriser Amazon Managed Service for Prometheus à accéder aux métriques. La rubrique suivante décrit comment configurer votre cluster.

Cross-account installation

Pour créer un scraper multicompte lorsque votre cluster Amazon EKS et l'espace de travail Amazon Managed Service for Prometheus se trouvent sur des comptes différents, procédez comme suit. Par exemple, vous avez un compte source `account_id_source` contenant le cluster Amazon EKS et un compte cible `account_id_target` contenant l'espace de travail Amazon Managed Service pour Prometheus.

Pour créer un scraper dans une configuration multi-comptes

1. Dans le compte source, créez un rôle `arn:aws:iam::account_id_source:role/Source` et ajoutez la politique de confiance suivante.

```
{
    "Effect": "Allow",
    "Principal": {
    "Service": [
        "scraper.aps.amazonaws.com"
     ]
    },
    "Action": "sts:AssumeRole",
    "Condition": {
        "ArnEquals": {
            "aws:SourceArn": "scraper_ARN"
        },
        "StringEquals": {
            "AWS:SourceAccount": "account_id"
        }
    }
}
```

2. Pour chaque combinaison de source (cluster Amazon EKS) et de cible (Amazon Managed Service for Prometheus workspace), vous devez créer un rôle `arn:aws:iam::account_id_target:role/Target` et ajoutez la politique de confiance suivante avec des autorisations pour AmazonPrometheusRemoteWriteAccess.

```
{
  "Effect": "Allow",
  "Principal": {
     "AWS": "arn:aws:iam::account_id_source:role/Source"
  },
  "Action": "sts:AssumeRole",
  "Condition": {
     "StringEquals": {
        "sts:ExternalId": "scraper_ARN"
      }
  }
}
```

3. Créez un grattoir à l'aide du `--role-configuration` option.

```
aws amp create-scraper \
  --source eksConfiguration="{clusterArn='arn:aws:eks:us-west-2:account-
id_source:cluster/xarw,subnetIds=[subnet-subnet-id]}" \
  --scrape-configuration configurationBlob=<base64-encoded-blob> \
  --destination ampConfiguration="{workspaceArn='arn:aws:aps:us-west-2:account-
id_target:workspace/ws-workspace-id'}"\
  --role-configuration '{"sourceRoleArn":"arn:aws:iam::account-id_source:role/
Source", "targetRoleArn":"arn:aws:iam::account-id_target:role/Target"}'
```

4. Validez la création du scraper.

```
aws amp list-scrapers
{
    "scrapers": [
        {
            "scraperId": "scraper-id",
            "arn": "arn:aws:aps:us-west-2:account_id_source:scraper/scraper-id",
            "roleArn": "arn:aws:iam::account_id_source:role/aws-service-role/
scraper.aps.amazonaws.com/
AWSServiceRoleForAmazonPrometheusScraperInternal_cc319052-41a3-4",
            "status": {
                "statusCode": "ACTIVE"
            },
            "createdAt": "2024-10-29T16:37:58.789000+00:00",
            "lastModifiedAt": "2024-10-29T16:55:17.085000+00:00",
            "tags": {},
            "source": {
                "eksConfiguration": {
                    "clusterArn": "arn:aws:eks:us-west-2:account_id_source:cluster/
xarw",
                    "securityGroupIds": [
                        "sg-security-group-id",
                        "sg-security-group-id"
                    ],
                    "subnetIds": [
                        "subnet-subnet_id"
                    ]
                }
```

```
            },
            "destination": {
                "ampConfiguration": {
                    "workspaceArn": "arn:aws:aps:us-
west-2:account_id_target:workspace/ws-workspace-id"
                }
            }
        }
    ]
}
```

Basculer entre RoleConfiguration et rôle lié aux services

Lorsque vous souhaitez revenir à un rôle lié à un service au lieu du `RoleConfiguration` pour écrire dans un espace de travail Amazon Managed Service for Prometheus, vous devez mettre à jour `UpdateScraper` et fournissez un espace de travail sur le même compte que le grattoir sans `RoleConfiguration`. Le `RoleConfiguration` sera supprimé du scraper et le rôle lié au service sera utilisé.

Lorsque vous changez d'espace de travail sur le même compte que le scraper et que vous souhaitez continuer à utiliser `RoleConfiguration`, vous devez à nouveau fournir le `RoleConfiguration` sur `UpdateScraper`.

Création d'un scraper pour les espaces de travail activé à l'aide de clés gérées par le client

Pour créer un scraper permettant d'ingérer des métriques dans un espace de travail Amazon Managed Service for Prometheus avec clés gérées par le client, utilisez le `--role-configuration` avec la source et la cible définies sur le même compte.

```
aws amp create-scraper \
  --source eksConfiguration="{clusterArn='arn:aws:eks:us-west-2:account-id:cluster/
xarw,subnetIds=[subnet-subnet_id]}" \
  --scrape-configuration configurationBlob=<base64-encoded-blob> \
  --destination ampConfiguration="{workspaceArn='arn:aws:aps:us-west-2:account-
id:workspace/ws-workspace-id'}"\
  --role-configuration '{"sourceRoleArn":"arn:aws:iam::account_id:role/Source",
 "targetRoleArn":"arn:aws:iam::account_id:role/Target"}'
```

Erreurs courantes lors de la création de grattoirs

Les problèmes les plus courants rencontrés lors de la tentative de création d'un nouveau scraper sont les suivants.

- Nécessaire AWS les ressources n'existent pas. Le groupe de sécurité, sous-réseaux, et Cluster Amazon EKS spécifié doit exister.
- Espace d'adressage IP insuffisant. Vous devez disposer d'au moins une adresse IP dans chaque sous-réseau que vous transmettez au `CreateScraper` API.

Configuration de votre cluster Amazon EKS

Votre cluster Amazon EKS doit être configuré pour permettre au scraper d'accéder aux métriques. Il existe deux options pour cette configuration :

- Utiliser Amazon EKS entrées d'accès pour fournir automatiquement à Amazon Managed Service l'accès des collecteurs Prometheus à votre cluster.
- Configurez manuellement votre cluster Amazon EKS pour le grattage géré des métriques.

Les rubriques suivantes décrivent chacun de ces éléments plus en détail.

Configurer Amazon EKS pour l'accès au scraper avec des entrées d'accès

L'utilisation des entrées d'accès pour Amazon EKS est le moyen le plus simple de permettre à Amazon Managed Service for Prometheus d'accéder aux métriques de votre cluster.

Le cluster Amazon EKS que vous êtes en train de récupérer doit être configuré pour autoriser l'authentification par API. Le mode d'authentification du cluster doit être défini sur `API` ou `API_AND_CONFIG_MAP`. Ceci est visible dans la console Amazon EKS sur le Configuration de l'accès onglet des détails du cluster. Pour plus d'informations, voir Autoriser les rôles IAM ou les utilisateurs à accéder à l'objet Kubernetes sur votre cluster Amazon EKS dans le Guide de l'utilisateur d'Amazon EKS.

Vous pouvez créer le scraper lors de la création du cluster ou après avoir créé le cluster :

- Lors de la création d'un cluster — Vous pouvez configurer cet accès lorsque vous créer un cluster Amazon EKS via la console Amazon EKS (suivez les instructions pour créer un scraper dans le cadre du cluster), et une politique de saisie d'accès sera automatiquement créée, permettant à Amazon Managed Service for Prometheus d'accéder aux métriques du cluster.

- Ajouter après la création d'un cluster — si votre cluster Amazon EKS existe déjà, définissez le mode d'authentification sur `API` ou `API_AND_CONFIG_MAP`, et tous les grattoirs que vous créez via l'API ou l'interface de ligne de commande Amazon Managed Service for Prometheus ou via la console Amazon EKS, la politique de saisie d'accès appropriée sera automatiquement créée pour vous, et les scrapers auront accès à votre cluster.

Politique d'entrée d'accès créée

Lorsque vous créez un scraper et que vous laissez Amazon Managed Service for Prometheus générer une politique de saisie d'accès pour vous, la politique suivante est générée. Pour plus d'informations sur les entrées d'accès, voir Autoriser les rôles IAM ou les utilisateurs à accéder à Kubernetes dans le Guide de l'utilisateur d'Amazon EKS.

```
{
    "rules": [
        {
            "effect": "allow",
            "apiGroups": [
                ""
            ],
            "resources": [
                "nodes",
                "nodes/proxy",
                "nodes/metrics",
                "services",
                "endpoints",
                "pods",
                "ingresses",
                "configmaps"
            ],
            "verbs": [
                "get",
                "list",
                "watch"
            ]
        },
        {
            "effect": "allow",
            "apiGroups": [
                "extensions",
                "networking.k8s.io"
            ],
```

```
            "resources": [
                "ingresses/status",
                "ingresses"
            ],
            "verbs": [
                "get",
                "list",
                "watch"
            ]
        },
        {
            "effect": "allow",
            "apiGroups": [
                "metrics.eks.amazonaws.com"
            ],
            "resources": [
                "kcm/metrics",
                "ksh/metrics"
            ],
            "verbs": [
                "get"
            ]
        },
        {
            "effect": "allow",
            "nonResourceURLs": [
                "/metrics"
            ],
            "verbs": [
                "get"
            ]
        }
    ]
}
```

Configuration manuelle d'Amazon EKS pour l'accès au scraper

Si vous préférez utiliser le `aws-auth ConfigMap` pour contrôler l'accès à votre cluster Kubernetes, vous pouvez toujours autoriser Amazon Managed Service for Prometheus scrapers à accéder à vos métriques. Les étapes suivantes permettront à Amazon Managed Service for Prometheus d'accéder aux métriques de votre cluster Amazon EKS.

> **Note**
>
> Pour plus d'informations sur `ConfigMap` et accédez aux entrées, voir Autoriser les rôles IAM ou les utilisateurs à accéder à Kubernetes dans le Guide de l'utilisateur d'Amazon EKS.

Cette procédure utilise `kubectl` et le AWS CLI. Pour plus d'informations sur l'installation de `kubectl`, consultez Installation de kubectl dans le Guide de l'utilisateur Amazon EKS.

Pour configurer manuellement votre cluster Amazon EKS pour le grattage géré des métriques

1. Créez un fichier appelé `clusterrole-binding.yml` avec le texte suivant :

```
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
  name: aps-collector-role
rules:
  - apiGroups: [""]
    resources: ["nodes", "nodes/proxy", "nodes/metrics", "services", "endpoints",
 "pods", "ingresses", "configmaps"]
    verbs: ["describe", "get", "list", "watch"]
  - apiGroups: ["extensions", "networking.k8s.io"]
    resources: ["ingresses/status", "ingresses"]
    verbs: ["describe", "get", "list", "watch"]
  - nonResourceURLs: ["/metrics"]
    verbs: ["get"]
  - apiGroups: ["metrics.eks.amazonaws.com"]
    resources: ["kcm/metrics", "ksh/metrics"]
    verbs: ["get"]
---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
  name: aps-collector-user-role-binding
subjects:
- kind: User
  name: aps-collector-user
  apiGroup: rbac.authorization.k8s.io
roleRef:
  kind: ClusterRole
  name: aps-collector-role
```

```
  apiGroup: rbac.authorization.k8s.io
```

2. Exécutez la commande suivante dans votre cluster :

```
kubectl apply -f clusterrole-binding.yml
```

Le lien et la règle du rôle du cluster sont alors créés. Cet exemple utilise `aps-collector-role` comme nom de rôle et `aps-collector-user` comme nom d'utilisateur.

3. La commande suivante vous donne des informations sur le grattoir avec l'ID *`scraper-id`*. Il s'agit du scraper que vous avez créé à l'aide de la commande de la section précédente.

```
aws amp describe-scraper --scraper-id scraper-id
```

4. À partir des résultats du`describe-scraper`, recherchez le `roleArn`. Son format est le suivant :

```
arn:aws:iam::account-id:role/aws-service-role/scraper.aps.amazonaws.com/
AWSServiceRoleForAmazonPrometheusScraper_unique-id
```

Amazon EKS nécessite un format différent pour cet ARN. Vous devez ajuster le format de l'ARN renvoyé pour l'utiliser à l'étape suivante. Modifiez-le pour qu'il corresponde au format suivant :

```
arn:aws:iam::account-id:role/AWSServiceRoleForAmazonPrometheusScraper_unique-id
```

Par exemple, l'ARN suivant :

```
arn:aws:iam::111122223333:role/aws-service-role/scraper.aps.amazonaws.com/
AWSServiceRoleForAmazonPrometheusScraper_1234abcd-56ef-7
```

Doit être réécrit comme suit :

```
arn:aws:iam::111122223333:role/
AWSServiceRoleForAmazonPrometheusScraper_1234abcd-56ef-7
```

5. Exécutez la commande suivante dans votre cluster, en utilisant le `roleArn` modifié de l'étape précédente, ainsi que le nom et la région du cluster.

```
eksctl create iamidentitymapping --cluster cluster-name --region region-id --
arn roleArn --username aps-collector-user
```

Le scraper peut ainsi accéder au cluster en utilisant le rôle et l'utilisateur que vous avez créés dans le fichier `clusterrole-binding.yml`.

Recherche et suppression des scrapers

Vous pouvez utiliser le AWS API ou AWS CLI pour répertorier les scrapers de votre compte ou pour les supprimer.

> **Note**
>
> Assurez-vous que vous utilisez la dernière version du AWS CLI ou SDK. La dernière version vous fournit les dernières fonctionnalités et fonctionnalités, ainsi que des mises à jour de sécurité. Vous pouvez également utiliser AWS CloudShell, qui fournit automatiquement une expérience de ligne de commande toujours à jour.

Pour afficher tous les scrapers de votre compte, utilisez l'opératon d'API ListScrapers.

Alternativement, avec AWS CLI, appelez :

```
aws amp list-scrapers --region aws-region
```

`ListScrapers` renvoie tous les scrapers de votre compte. Par exemple :

```
{
    "scrapers": [
        {
            "scraperId": "s-1234abcd-56ef-7890-abcd-1234ef567890",
            "arn": "arn:aws:aps:us-west-2:123456789012:scraper/s-1234abcd-56ef-7890-
abcd-1234ef567890",
            "roleArn": "arn:aws:iam::123456789012:role/aws-service-role/
AWSServiceRoleForAmazonPrometheusScraper_1234abcd-2931",
            "status": {
                "statusCode": "DELETING"
            },
            "createdAt": "2023-10-12T15:22:19.014000-07:00",
```

```
            "lastModifiedAt": "2023-10-12T15:55:43.487000-07:00",
            "tags": {},
            "source": {
                "eksConfiguration": {
                    "clusterArn": "arn:aws:eks:us-west-2:123456789012:cluster/my-
cluster",
                    "securityGroupIds": [
                        "sg-1234abcd5678ef90"
                    ],
                    "subnetIds": [
                        "subnet-abcd1234ef567890",
                        "subnet-1234abcd5678ab90"
                    ]
                }
            },
            "destination": {
                "ampConfiguration": {
                    "workspaceArn": "arn:aws:aps:us-west-2:123456789012:workspace/
ws-1234abcd-5678-ef90-ab12-cdef3456a78"
                }
            }
        }
    ]
}
```

Pour supprimer un scraper, recherchez le `scraperId` du scraper que vous souhaitez supprimer à l'aide de l'opération `ListScrapers`, puis utilisez l'opération DeleteScraper pour le supprimer.

Alternativement, avec AWS CLI, appelez :

```
aws amp delete-scraper --scraper-id scraperId
```

Configuration du scraper

Vous pouvez contrôler la façon dont votre scraper découvre et collecte les métriques à l'aide d'un Prometheus-compatible configuration du grattoir. Par exemple, vous pouvez modifier l'intervalle d'envoi des métriques à l'espace de travail. Vous pouvez également utiliser le réétiquetage pour réécrire dynamiquement les étiquettes d'une métrique. La configuration du scraper est un fichier YAML qui fait partie de la définition du scraper.

Lorsqu'un nouveau scraper est créé, vous spécifiez une configuration en fournissant un fichier YAML codé en base64 dans l'appel d'API. Vous pouvez télécharger un fichier de configuration générale

avec l'opération `GetDefaultScraperConfiguration` dans l'API Amazon Managed Service for Prometheus.

Pour modifier la configuration d'un scraper, vous pouvez utiliser `UpdateScraper` opération. Si vous devez mettre à jour la source des métriques (par exemple, vers un autre cluster Amazon EKS), vous devez supprimer le scraper et le recréer avec la nouvelle source.

Configuration prise en charge

Pour plus d'informations sur le format de configuration du scraper, y compris une ventilation détaillée des valeurs possibles, voir Configuration dans la documentation Prometheus. Les options de configuration globale et les options `<scrape_config>` décrivent les options les plus fréquemment requises.

Amazon EKS étant le seul service pris en charge, la seule configuration de découverte de service (`<*_sd_config>`) pris en charge est le `<kubernetes_sd_config>`.

La liste complète des sections de configuration autorisées :

- `<global>`
- `<scrape_config>`
- `<static_config>`
- `<relabel_config>`
- `<metric_relabel_configs>`
- `<kubernetes_sd_config>`

Les limites de ces sections sont répertoriées après l'exemple de fichier de configuration.

Exemple de fichier de configuration

Voici un exemple de fichier de configuration YAML avec un intervalle de récupération de 30 secondes. Cet exemple inclut la prise en charge des métriques du serveur d'API kube, ainsi que des métriques kube-controller-manager et kube-scheduler. Pour en savoir plus, consultez Récupérer les métriques brutes du plan de contrôle au format Prometheus dans le Guide de l'utilisateur Amazon EKS.

```
global:
   scrape_interval: 30s
   external_labels:
```

```
     clusterArn: apiserver-test-2
scrape_configs:
  - job_name: pod_exporter
    kubernetes_sd_configs:
      - role: pod
  - job_name: cadvisor
    scheme: https
    authorization:
      type: Bearer
      credentials_file: /var/run/secrets/kubernetes.io/serviceaccount/token
    kubernetes_sd_configs:
      - role: node
    relabel_configs:
      - action: labelmap
        regex: __meta_kubernetes_node_label_(.+)
      - replacement: kubernetes.default.svc:443
        target_label: __address__
      - source_labels: [__meta_kubernetes_node_name]
        regex: (.+)
        target_label: __metrics_path__
        replacement: /api/v1/nodes/$1/proxy/metrics/cadvisor
  # apiserver metrics
  - scheme: https
    authorization:
      type: Bearer
      credentials_file: /var/run/secrets/kubernetes.io/serviceaccount/token
    job_name: kubernetes-apiservers
    kubernetes_sd_configs:
    - role: endpoints
    relabel_configs:
    - action: keep
      regex: default;kubernetes;https
      source_labels:
      - __meta_kubernetes_namespace
      - __meta_kubernetes_service_name
      - __meta_kubernetes_endpoint_port_name
  # kube proxy metrics
  - job_name: kube-proxy
    honor_labels: true
    kubernetes_sd_configs:
    - role: pod
    relabel_configs:
    - action: keep
      source_labels:
```

```
      - __meta_kubernetes_namespace
      - __meta_kubernetes_pod_name
      separator: '/'
      regex: 'kube-system/kube-proxy.+'
    - source_labels:
      - __address__
      action: replace
      target_label: __address__
      regex: '(.+?)(:\d+)?'
      replacement: $1:10249
  # Scheduler metrics
  - job_name: 'ksh-metrics'
    kubernetes_sd_configs:
    - role: endpoints
    metrics_path: /apis/metrics.eks.amazonaws.com/v1/ksh/container/metrics
    scheme: https
    bearer_token_file: /var/run/secrets/kubernetes.io/serviceaccount/token
    relabel_configs:
    - source_labels:
      - __meta_kubernetes_namespace
      - __meta_kubernetes_service_name
      - __meta_kubernetes_endpoint_port_name
      action: keep
      regex: default;kubernetes;https
  # Controller Manager metrics
  - job_name: 'kcm-metrics'
    kubernetes_sd_configs:
    - role: endpoints
    metrics_path: /apis/metrics.eks.amazonaws.com/v1/kcm/container/metrics
    scheme: https
    bearer_token_file: /var/run/secrets/kubernetes.io/serviceaccount/token
    relabel_configs:
    - source_labels:
      - __meta_kubernetes_namespace
      - __meta_kubernetes_service_name
      - __meta_kubernetes_endpoint_port_name
      action: keep
      regex: default;kubernetes;https
```

Les limites suivantes sont spécifiques à AWS collectionneurs gérés :

- Intervalle de scrape : la configuration du scraper ne peut pas spécifier un intervalle de scrape inférieur à 30 secondes.

- Cibles : les cibles de `static_config` doivent être spécifiées sous la forme d'adresses IP.
- Résolution DNS — En ce qui concerne le nom de la cible, le seul nom de serveur reconnu dans cette configuration est le serveur API Kubernetes, `kubernetes.default.svc`. Tous les autres noms de machines doivent être spécifiés par adresse IP.
- Autorisation — Omettre si aucune autorisation n'est requise. Si nécessaire, l'autorisation doit être `Bearer`, et doit pointer vers le fichier `/var/run/secrets/kubernetes.io/serviceaccount/token`. En d'autres termes, si elle est utilisée, la section d'autorisation doit ressembler à ce qui suit :

```
authorization:
  type: Bearer
  credentials_file: /var/run/secrets/kubernetes.io/serviceaccount/token
```

> **Note**
>
> `type: Bearer` est la valeur par défaut, elle peut donc être omise.

Résolution des erreurs de configuration du scraper

Les collecteurs Amazon Managed Service for Prometheus reconnaissent et collectent automatiquement les métriques. Mais comment résoudre le problème lorsque vous ne voyez pas une métrique que vous vous attendiez à voir dans votre espace de travail Amazon Managed Service for Prometheus ?

> **Important**
>
> Vérifiez que l'accès privé pour votre cluster Amazon EKS est activé. Pour plus d'informations, voir Point de terminaison privé du cluster dans le Guide de l'utilisateur d'Amazon EKS.

La métrique up est un outil utile. Pour chaque point de terminaison reconnu par un collecteur Amazon Managed Service for Prometheus, ce dernier envoie automatiquement cette métrique. Il existe trois états de cette métrique qui peuvent vous aider à résoudre les problèmes qui se produisent dans le collecteur.

- up n'est pas présent – Si aucune métrique up n'est présente pour un point de terminaison, cela signifie que le collecteur n'a pas pu trouver le point de terminaison.

Si vous êtes certain que le point de terminaison existe, le collecteur peut ne pas le trouver pour plusieurs raisons.

- Il se peut que vous deviez ajuster la configuration du scrape. La découverte `relabel_config` pourrait avoir besoin d'être ajusté.
- Il se peut qu'il y ait un problème avec `role` utilisé pour la découverte.
- L'Amazon VPC utilisé par le cluster Amazon EKS n'a peut-être pas DNS activé, ce qui empêcherait le collecteur de trouver le point final.

- up est présent, mais vaut toujours 0 — Si up est présent, mais 0, alors le collecteur est capable de découvrir le point de terminaison, mais n'en trouve aucun Prometheus-compatible métriques.

  Dans ce cas, vous pouvez essayer d'utiliser une commande `curl` directement sur le point de terminaison. Vous pouvez vérifier que les informations sont correctes, par exemple, le protocole (`http` ou `https`), le terminal ou le port que vous utilisez. Vous pouvez également vérifier que le terminal répond avec un message valide `200` réponse, et suit le format Prometheus. Enfin, le corps de la réponse ne peut pas être plus grand que la taille maximale autorisée. (Pour les limites de AWS collecteurs gérés, voir la section suivante.)

- up est présent et supérieur à 0 – Si up est présent et supérieur à 0, cela signifie que les métriques sont envoyées à Amazon Managed Service for Prometheus.

  Assurez-vous de rechercher les bonnes métriques dans Amazon Managed Service for Prometheus (ou dans votre autre tableau de bord, par exemple Amazon Managed Grafana). Vous pouvez à nouveau utiliser curl pour vérifier les données attendues sur votre point de terminaison `/metrics`. Vérifiez également que vous n'avez pas dépassé les autres limites, telles que le nombre de points de terminaison par scraper. Vous pouvez vérifier le nombre de points de terminaison des métriques extraits en vérifiant le nombre de up métriques, en utilisant `count(up)`.

Limitations du scraper

Les scrapers entièrement gérés fournis par Amazon Managed Service for Prometheus sont soumis à quelques limitations.

- Région – Votre cluster EKS, votre scraper géré et votre espace de travail Amazon Managed Service for Prometheus doivent tous se trouver dans la même région AWS .
- Collecteurs – Vous pouvez disposer d'un maximum de 10 scrapers Amazon Managed Service for Prometheus par région et par compte.

> **Note**
>
> Vous pouvez demander une augmentation de cette limite en demandant une augmentation de quota.

- Réponse aux métriques – Le corps d'une réponse provenant d'une demande de point de terminaison `/metrics` ne peut pas dépasser 50 mégaoctets (Mo).
- Points de terminaison par scraper – Un scraper peut collecter jusqu'à 30 000 points de terminaison `/metrics`.
- Intervalle de scrape : la configuration du scraper ne peut pas spécifier un intervalle de scrape inférieur à 30 secondes.

Configurer des collecteurs Prometheus gérés pour Amazon MSK

Pour utiliser un collecteur Amazon Managed Service for Prometheus, vous devez créer un scraper qui découvre et extrait des métriques dans votre cluster Amazon Managed Streaming pour Apache Kafka. Vous pouvez également créer un scraper qui s'intègre à Amazon Elastic Kubernetes Service ou à Amazon OpenSearch Service. Pour plus d'informations, voir Intégrez Amazon EKS et Intégrez Amazon OpenSearch Service.

Pour obtenir des instructions sur la collecte du décalage par groupe de consommateurs, les compensations validées et l'adhésion à un exportateur Kafka géré, voir Collectez des statistiques sur les groupes de consommateurs Amazon MSK. Cette procédure utilise AWS Authentification IAM (Identity and Access Management) et ne nécessite pas de surveillance ouverte. La procédure suivante collecte les métriques JMX et Node Exporter à partir d'Open Monitoring.

Créer un scraper

Un collecteur Amazon Managed Service for Prometheus consiste en un scraper qui découvre et collecte des métriques à partir d'un cluster Amazon MSK. Amazon Managed Service for Prometheus gère le scraper pour vous, vous offrant ainsi l'évolutivité, la sécurité et la fiabilité dont vous avez besoin, sans avoir à gérer vous-même les instances, les agents ou les scrapers.

Vous pouvez créer un grattoir à l'aide du AWS API ou AWS CLI comme décrit dans les procédures suivantes.

Il existe quelques prérequis pour créer votre propre scraper :

- Vous devez avoir créé un cluster Amazon MSK.
- Configurez le groupe de sécurité de votre cluster Amazon MSK pour autoriser le trafic entrant sur les ports 11001 (Exportateur JMX) et 11002 (exportateur de nœuds) dans votre Amazon VPC, car le scraper a besoin d'accéder à ces enregistrements DNS pour collecter des métriques Prometheus.
- L'Amazon VPC dans lequel se trouve le cluster Amazon MSK doit avoir DNS activé.

> **Note**
>
> Le cluster sera associé au scraper par son nom de ressource Amazon (ARN). Si vous supprimez un cluster, puis que vous en créez un nouveau avec le même nom, l'ARN sera réutilisé pour le nouveau cluster. Pour cette raison, le scraper tentera de collecter des métriques pour le nouveau cluster. Vous supprimer les grattoirs séparément de la suppression du cluster.

To create a scraper using the AWS API

Utilisez le `CreateScraper` Fonctionnement de l'API pour créer un scraper avec AWS API. L'exemple suivant crée un scraper dans la région de l'Est des États-Unis (Virginie du Nord). Remplacez le *`example`* du contenu avec les informations de votre cluster Amazon MSK et fournissez la configuration de votre scraper.

> **Note**
>
> Configurez le groupe de sécurité et les sous-réseaux en fonction de votre cluster cible. Incluez au moins deux sous-réseaux dans deux zones de disponibilité.

```
                POST /scrapers HTTP/1.1
Content-Length: 415
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: aws-cli/1.18.147 Python/2.7.18 Linux/5.4.58-37.125.amzn2int.x86_64
 botocore/1.18.6

{
```

```
    "alias": "myScraper",
    "destination":  {
        "ampConfiguration": {
            "workspaceArn": "arn:aws:aps:us-east-1:123456789012:workspace/ws-
workspace-id"
        }
    },
    "source": {
        "vpcConfiguration": {
            "securityGroupIds": ["sg-security-group-id"],
            "subnetIds": ["subnet-subnet-id-1", "subnet-subnet-id-2"]
        }
    },
    "scrapeConfiguration": {
        "configurationBlob": base64-encoded-blob
    }
}
```

Dans l'exemple, `scrapeConfiguration` Le paramètre nécessite un fichier YAML de configuration Prometheus codé en base64 qui spécifie les enregistrements DNS du cluster MSK.

Chaque enregistrement DNS représente un point de terminaison de courtier dans une zone de disponibilité spécifique, permettant aux clients de se connecter à des courtiers répartis sur les zones de disponibilité de votre choix pour une haute disponibilité.

Le nombre d'enregistrements DNS dans les propriétés de votre cluster MSK correspond au nombre de nœuds courtiers et de zones de disponibilité dans la configuration de votre cluster :

- Configuration par défaut — 3 nœuds de courtage répartis sur 3 AZ = 3 enregistrements DNS
- Configuration personnalisée — 2 nœuds de courtage sur 2 AZ = 2 enregistrements DNS

Pour obtenir les enregistrements DNS de votre cluster MSK, ouvrez la console MSK à l'adresse https://console.aws.amazon.com/msk/home? region=us-east-1#/accueil/. Accédez à votre cluster MSK. Choisissez Propriétés, Courtiers, et Points finaux.

Deux options s'offrent à vous pour configurer Prometheus afin de récupérer les métriques de votre cluster MSK :

1. Cluster-level Résolution DNS (recommandée) — Utilisez le nom DNS de base du cluster pour découvrir automatiquement tous les courtiers. Si le point de terminaison de votre courtier

est `b-1.clusterName.xxx.xxx.xxx`, utiliser `clusterName.xxx.xxx.xxx` en tant qu'enregistrement DNS. Cela permet à Prometheus de scraper automatiquement tous les courtiers du cluster.

Points de terminaison individuels pour les courtiers — Spécifiez chaque point de terminaison de courtier individuellement pour un contrôle granulaire. Utilisez les identifiants de courtier complets (b-1, b-2) dans votre configuration. Par exemple :

```
dns_sd_configs:
  - names:
    - b-1.clusterName.xxx.xxx.xxx
    - b-2.clusterName.xxx.xxx.xxx
    - b-3.clusterName.xxx.xxx.xxx
```

> **Note**
>
> Remplacer `clusterName.xxx.xxx.xxx` avec le point de terminaison actuel de votre cluster MSK à partir du AWS Console.

Pour plus d'informations, voir <dns_sd_config> dans le Prométhée documentation.

Voici un exemple de fichier de configuration du scraper :

```
global:
  scrape_interval: 30s
  external_labels:
    clusterArn: msk-test-1

scrape_configs:
  - job_name: msk-jmx
    scheme: http
    metrics_path: /metrics
    scrape_timeout: 10s
    dns_sd_configs:
      - names:
          - dns-record-1
          - dns-record-2
          - dns-record-3
        type: A
```

```
        port: 11001
    relabel_configs:
      - source_labels: [__meta_dns_name]
        target_label: broker_dns
      - source_labels: [__address__]
        target_label: instance
        regex: '(.*)'
        replacement: '${1}'

  - job_name: msk-node
    scheme: http
    metrics_path: /metrics
    scrape_timeout: 10s
    dns_sd_configs:
      - names:
          - dns-record-1
          - dns-record-2
          - dns-record-3
        type: A
        port: 11002
    relabel_configs:
      - source_labels: [__meta_dns_name]
        target_label: broker_dns
      - source_labels: [__address__]
        target_label: instance
        regex: '(.*)'
        replacement: '${1}'
```

Exécutez l'une des commandes suivantes pour convertir le fichier YAML en base64. Vous pouvez également utiliser n'importe quel convertisseur base64 en ligne pour convertir le fichier.

Example Linux/macOS

```
echo -n scraper config updated with dns records | base64
```

Example Fenêtres PowerShell

```
[Convert]::ToBase64String([System.Text.Encoding]::UTF8.GetBytes(scraper config
 updated with dns records))
```

To create a scraper using the AWS CLI

Utilisez le `create-scraper` commande pour créer un grattoir à l'aide de AWS Command Line Interface. L'exemple suivant crée un scraper dans la région de l'Est des États-Unis (Virginie du Nord). Remplacez le *example* du contenu avec les informations de votre cluster Amazon MSK et fournissez la configuration de votre scraper.

> **Note**
>
> Configurez le groupe de sécurité et les sous-réseaux en fonction de votre cluster cible. Incluez au moins deux sous-réseaux dans deux zones de disponibilité.

```
aws amp create-scraper \
 --source vpcConfiguration="{securityGroupIds=['sg-security-group-
id'],subnetIds=['subnet-subnet-id-1', 'subnet-subnet-id-2']}" \
--scrape-configuration configurationBlob=base64-encoded-blob \
 --destination ampConfiguration="{workspaceArn='arn:aws:aps:us-
west-2:123456789012:workspace/ws-workspace-id'}"
```

- Vous trouverez ci-dessous une liste complète des opérations de grattage que vous pouvez utiliser avec AWS API :

  Créez un scraper avec l'opération d'API CreateScraper.
- Répertoriez vos scrapers existants avec l'opération d'API ListScrapers.
- Mettez à jour l'alias, la configuration ou la destination d'un scraper avec UpdateScraper Fonctionnement de l'API.
- Supprimez un scraper avec l'opération d'API DeleteScraper.
- Obtenez plus de détails sur un scraper avec l'opération d'API DescribeScraper.

Cross-account installation

Pour créer un scraper dans une configuration multi-comptes lorsque le cluster Amazon MSK à partir duquel vous souhaitez collecter des métriques se trouve dans un compte différent de celui du collecteur Amazon Managed Service for Prometheus, suivez la procédure ci-dessous.

Par exemple, lorsque vous avez deux comptes, le premier compte source `account_id_source` où se trouve l'Amazon MSK, et un deuxième compte cible `account_id_target` où se trouve l'espace de travail Amazon Managed Service for Prometheus.

Pour créer un scraper dans une configuration multi-comptes

1. Dans le compte source, créez un rôle `arn:aws:iam::111122223333:role/Source` et ajoutez la politique de confiance suivante.

```
{
    "Effect": "Allow",
    "Principal": {
    "Service": [
        "scraper.aps.amazonaws.com"
     ]
    },
    "Action": "sts:AssumeRole",
    "Condition": {
        "ArnEquals": {
            "aws:SourceArn": "arn:aws:aps:aws-region:111122223333:scraper/scraper-id"
        },
        "StringEquals": {
            "AWS:SourceAccount": "111122223333"
        }
    }
}
```

2. Pour chaque combinaison de source (cluster Amazon MSK) et de cible (Amazon Managed Service for Prometheus workspace), vous devez créer un rôle `arn:aws:iam::444455556666:role/Target` et ajoutez la politique de confiance suivante avec des autorisations pour AmazonPrometheusRemoteWriteAccess.

```
{
  "Effect": "Allow",
  "Principal": {
     "AWS": "arn:aws:iam::111122223333:role/Source"
  },
  "Action": "sts:AssumeRole",
  "Condition": {
```

```
     "StringEquals": {
        "sts:ExternalId": "arn:aws:aps:aws-region:111122223333:scraper/scraper-id"
      }
  }
}
```

3. Créez un grattoir à l'aide du `--role-configuration` option.

```
aws amp create-scraper \ --source vpcConfiguration="{subnetIds=[subnet-
subnet-id], "securityGroupIds": ["sg-security-group-id"]}" \ --
scrape-configuration configurationBlob=<base64-encoded-blob> \
 --destination ampConfiguration="{workspaceArn='arn:aws:aps:aws-
region:444455556666:workspace/ws-workspace-id'}"\ --role-configuration
 '{"sourceRoleArn":"arn:aws:iam::111122223333:role/Source",
 "targetRoleArn":"arn:aws:iam::444455556666:role/Target"}'
```

4. Validez la création du scraper.

```
aws amp list-scrapers
{
    "scrapers": [
        {
            "scraperId": "s-example123456789abcdef0",
            "arn": "arn:aws:aps:aws-region:111122223333:scraper/s-
example123456789abcdef0": "arn:aws:iam::111122223333:role/Source",
            "status": "ACTIVE",
            "creationTime": "2025-10-27T18:45:00.000Z",
            "lastModificationTime": "2025-10-27T18:50:00.000Z",
            "tags": {},
            "statusReason": "Scraper is running successfully",
            "source": {
                "vpcConfiguration": {
                    "subnetIds": ["subnet-subnet-id"],
                    "securityGroupIds": ["sg-security-group-id"]
                }
            },
            "destination": {
                "ampConfiguration": {
                    "workspaceArn": "arn:aws:aps:aws-region:444455556666:workspace/
ws-workspace-id'"
                }
```

```
            },
            "scrapeConfiguration": {
                "configurationBlob": "<base64-encoded-blob>"
            }
        }
    ]
}
```

Basculer entre RoleConfiguration et rôle lié aux services

Lorsque vous souhaitez revenir à un rôle lié à un service au lieu du `RoleConfiguration` pour écrire dans un espace de travail Amazon Managed Service for Prometheus, vous devez mettre à jour `UpdateScraper` et fournissez un espace de travail sur le même compte que le grattoir sans `RoleConfiguration`. Le `RoleConfiguration` sera supprimé du scraper et le rôle lié au service sera utilisé.

Lorsque vous changez d'espace de travail sur le même compte que le scraper et que vous souhaitez continuer à utiliser `RoleConfiguration`, vous devez à nouveau fournir le `RoleConfiguration` sur `UpdateScraper`.

Recherche et suppression des scrapers

Vous pouvez utiliser le AWS API ou AWS CLI pour répertorier les scrapers de votre compte ou pour les supprimer.

> **Note**
>
> Assurez-vous que vous utilisez la dernière version du AWS CLI ou SDK. La dernière version vous fournit les dernières fonctionnalités et fonctionnalités, ainsi que des mises à jour de sécurité. Vous pouvez également utiliser AWS CloudShell, qui fournit automatiquement une expérience de ligne de commande toujours à jour.

Pour afficher tous les scrapers de votre compte, utilisez l'opératon d'API ListScrapers.

Alternativement, avec AWS CLI, appelez :

```
aws amp list-scrapers
```

`ListScrapers` renvoie tous les scrapers de votre compte. Par exemple :

```
{
    "scrapers": [
        {
            "scraperId": "s-1234abcd-56ef-7890-abcd-1234ef567890",
            "arn": "arn:aws:aps:aws-region:123456789012:scraper/s-1234abcd-56ef-7890-
abcd-1234ef567890",
            "roleArn": "arn:aws:iam::123456789012:role/aws-service-role/
AWSServiceRoleForAmazonPrometheusScraper_1234abcd-2931",
            "status": {
                "statusCode": "DELETING"
            },
            "createdAt": "2023-10-12T15:22:19.014000-07:00",
            "lastModifiedAt": "2023-10-12T15:55:43.487000-07:00",
            "tags": {},
            "source": {
                "vpcConfiguration": {
                   "securityGroupIds": [
                        "sg-1234abcd5678ef90"
                    ],
                    "subnetIds": [
                        "subnet-abcd1234ef567890",
                        "subnet-1234abcd5678ab90"
                    ]
                }
            },
            "destination": {
                "ampConfiguration": {
                    "workspaceArn": "arn:aws:aps:aws-region:123456789012:workspace/
ws-1234abcd-5678-ef90-ab12-cdef3456a78"
                }
            }
        }
    ]
}
```

Pour supprimer un scraper, recherchez le `scraperId` du scraper que vous souhaitez supprimer à l'aide de l'opération `ListScrapers`, puis utilisez l'opération DeleteScraper pour le supprimer.

Alternativement, avec AWS CLI, appelez :

```
aws amp delete-scraper --scraper-id scraperId
```

Statistiques collectées auprès d'Amazon MSK

Lorsque vous intégrez Amazon MSK, le collecteur Amazon Managed Service for Prometheus extrait automatiquement les métriques suivantes :

Métriques : jobs jmx_exporter et pod_exporter

Métrique	Description/Objectif
jmx_config_reload_failure_total	Nombre total de fois où l'exportateur JMX n'a pas réussi à recharger son fichier de configura tion.
jmx_scrape_duration_seconds	Temps nécessaire pour extraire les métriques JMX en secondes pour le cycle de collecte en cours.
erreur jmx_scrape_error	Indique si une erreur s'est produite lors de l'extraction des métriques JMX (1 = erreur, 0 = succès).
java_lang_Memory_HeapMemoryUsage_used	Quantité de mémoire de tas (en octets) actuellement utilisée par la JVM.
java_lang_Memory_HeapMemoryUsage_max	Quantité maximale de mémoire (en octets) pouvant être utilisée pour la gestion de la mémoire.
java_lang_Memory_NonHeapMemoryUsage_ used	Quantité de mémoire hors tas (en octets) actuellement utilisée par la JVM.
kafka_cluster_Partition_Value	État ou valeur actuel lié aux partitions du cluster Kafka, ventilé par ID de partition et par sujet.

Métrique	Description/Objectif
Kafka_Consumer_Consumer_Coordinator_Metrics_Assigned_Partitions	Nombre de partitions actuellement attribuées à ce consommateur.
kafka_consumer_consumer_coordinator_metrics_commit_latency_avg	Temps moyen nécessaire pour valider les décalages en millisecondes.
kafka_consumer_consumer_coordinator_metrics_commit_rate	Nombre de validations de décalage par seconde.
kafka_consumer_consumer_coordinator_metrics_failed_rebalance_total	Nombre total d'échecs de rééquilibrage des groupes de consommateurs.
kafka_consumer_consumer_coordinator_metrics_last_heartbeat_seconds_ago	Nombre de secondes écoulées depuis que le dernier battement de cœur a été envoyé au coordinateur.
kafka_consumer_consumer_coordinator_metrics_rebalance_latency_avg	Temps moyen nécessaire au rééquilibrage des groupes de consommateurs en millisecondes.
kafka_consumer_consumer_coordinator_metrics_rebalance_total	Nombre total de rééquilibres de groupes de consommateurs.
kafka_consumer_consumer_fetch_manager_metrics_bytes_consumed_rate	Nombre moyen d'octets consommés par seconde par le consommateur.
kafka_consumer_consumer_fetch_manager_metrics_fetch_latency_avg	Durée moyenne d'une demande de récupération en millisecondes.
kafka_consumer_consumer_fetch_manager_metrics_fetch_rate	Nombre de demandes de récupération par seconde.
kafka_consumer_consumer_fetch_manager_metrics_records_consumed_rate	Nombre moyen d'enregistrements consommés par seconde.
kafka_consumer_consumer_fetch_manager_metrics_records_lag_max	Latence maximale en termes de nombre d'enregistrements pour n'importe quelle partition de ce consommateur.

Métrique	Description/Objectif
kafka_consumer_consumer_metrics_connection_count	Nombre actuel de connexions actives.
kafka_consumer_consumer_metrics_incoming_byte_rate	Nombre moyen d'octets reçus par seconde en provenance de tous les serveurs.
kafka_consumer_consumer_metrics_last_poll_seconds_ago	Nombre de secondes écoulées depuis le dernier appel pour le sondage auprès des consommateurs ().
kafka_consumer_consumer_metrics_request_rate	Nombre de demandes envoyées par seconde.
Kafka_Consumer_Consumer_Metrics_Response_Rate	Nombre de réponses reçues par seconde.
group_consommateurs_kafka_ConsumerLagMetrics_Valeur	Valeur de décalage actuelle des consommateurs pour un groupe de consommateurs, indiquant le retard par rapport au consommateur.
contrôleur_kafka_KafkaController_Valeur	État actuel ou valeur du contrôleur Kafka (1 = contrôleur actif, 0 = inactif).
contrôleur_kafka_ControllerEventManager_Compter	Nombre total d'événements de contrôleur traités.
contrôleur_kafka_ControllerEventManager_Moyenne	Temps moyen (moyen) nécessaire pour traiter les événements du contrôleur.
contrôleur_kafka_ControllerStats_MeanRate	Taux moyen d'opérations statistiques du contrôleur par seconde.
group_coordinateur_kafka_GroupMetadataManager_Valeur	État actuel ou valeur du gestionnaire de métadonnées de groupe pour les groupes de consommateurs.

Métrique	Description/Objectif
kafka_log_LogFlushStats_Compter	Nombre total d'opérations de rinçage des journaux.
kafka_log_LogFlushStats_Moyenne	Temps moyen (moyen) nécessaire aux opérations de vidange des journaux.
kafka_log_LogFlushStats_MeanRate	Taux moyen d'opérations de rinçage des journaux par seconde.
réseau_kafka_RequestMetrics_Compter	Nombre total de demandes réseau traitées.
réseau_kafka_RequestMetrics_Moyenne	Temps moyen (moyen) nécessaire au traitement des requêtes réseau.
réseau_kafka_RequestMetrics_MeanRate	Taux moyen de requêtes réseau par seconde.
réseau_kafka_Acceptor_MeanRate	Taux moyen de connexions acceptées par seconde.
serveur_kafka_Fetch_queue_size	Taille actuelle de la file d'attente des demandes de récupération.
serveur_kafka_Produce_queue_size	Taille actuelle de la file d'attente des demandes de production.
serveur_kafka_Request_queue_size	Taille actuelle de la file d'attente générale des demandes.
serveur_kafka_BrokerTopicMetrics_Compter	Nombre total d'opérations thématiques effectuées par les courtiers (messages) in/out, octets in/out).
serveur_kafka_BrokerTopicMetrics_MeanRate	Taux moyen d'opérations thématiques effectuées par les courtiers par seconde.
serveur_kafka_BrokerTopicMetrics_One MinuteRate	One-minute taux moyen mobile des opérations thématiques des courtiers.

Métrique	Description/Objectif
serveur_kafka_DelayedOperationPurgatory_Valeur	Nombre actuel d'opérations retardées au purgatoire (en attente d'achèvement).
serveur_kafka_DelayedFetchMetrics_MeanRate	Taux moyen d'opérations de récupération différées par seconde.
serveur_kafka_FetcherLagMetrics_Valeur	Valeur de décalage actuelle pour les threads de récupération des répliques (distance par rapport au leader).
serveur_kafka_FetcherStats_MeanRate	Taux moyen d'opérations de récupération par seconde.
serveur_kafka_ReplicaManager_Valeur	État actuel ou valeur du gestionnaire de répliques.
serveur_kafka_ReplicaManager_MeanRate	Taux moyen d'opérations du gestionnaire de répliques par seconde.
serveur_kafka_LeaderReplication_byte_rate	Taux d'octets répliqués par seconde pour les partitions où ce broker est le leader.
kafka_server_group_coordinator_metrics_group_completed_rebalance_count	Nombre total de rééquilibres de groupes de consommateurs achevés.
kafka_server_group_coordinator_metrics_offset_commit_count	Nombre total d'opérations de validation de compensation.
kafka_server_group_coordinator_metrics_offset_commit_rate	Taux d'opérations de validation de décalage par seconde.
kafka_server_socket_server_metrics_connection_count	Nombre actuel de connexions actives.
kafka_server_socket_server_metrics_connection_creation_rate	Taux de création de nouvelles connexions par seconde.

Métrique	Description/Objectif
kafka_server_socket_server_metrics_connection_close_rate	Taux de fermetures de connexions par seconde.
kafka_server_socket_server_metrics_failed_authentication_total	Nombre total de tentatives d'authentification échouées.
kafka_server_socket_server_metrics_incoming_byte_rate	Taux d'octets entrants par seconde.
kafka_server_socket_server_metrics_outgoing_byte_rate	Taux d'octets sortants par seconde.
kafka_server_socket_server_metrics_request_rate	Taux de demandes par seconde.
kafka_server_socket_server_metrics_response_rate	Taux de réponses par seconde.
kafka_server_socket_server_metrics_network_io_rate	Débit du réseau I/O opérations par seconde.
kafka_server_socket_server_metrics_io_ratio	Fraction du temps passé I/O opérations.
kafka_server_controller_channel_metrics_connection_count	Nombre actuel de connexions actives pour les canaux du contrôleur.
kafka_server_controller_channel_metrics_incoming_byte_rate	Taux d'octets entrants par seconde pour les canaux du contrôleur.
kafka_server_controller_channel_metrics_outgoing_byte_rate	Taux d'octets sortants par seconde pour les canaux du contrôleur.
kafka_server_controller_channel_metrics_request_rate	Taux de demandes par seconde pour les canaux du contrôleur.
kafka_server_replica_fetcher_metrics_connection_count	Nombre actuel de connexions actives pour le récupérateur de répliques.

Métrique	Description/Objectif
kafka_server_replica_fetcher_metrics_incoming_byte_rate	Taux d'octets entrants par seconde pour le récupérateur de répliques.
kafka_server_replica_fetcher_metrics_request_rate	Taux de demandes par seconde pour le récupérateur de répliques.
kafka_server_replica_fetcher_metrics_failed_authentication_total	Nombre total de tentatives d'authentification échouées pour le récupérateur de répliques.
serveur_kafka_ZooKeeperClientMetrics_Compter	Nombre total de ZooKeeper opérations avec les clients.
serveur_kafka_ZooKeeperClientMetrics_Moyenne	Latence moyenne de ZooKeeper opérations avec les clients.
serveur_kafka_KafkaServer_Valeur	État actuel ou valeur du serveur Kafka (indique généralement que le serveur est en cours d'exécution).
node_cpu_seconds_total	Nombre total de secondes passées par les processeurs dans chaque mode (utilisateur, système, inactif, etc.), ventilé par processeur et par mode.
node_disk_read_bytes_total	Nombre total d'octets lus avec succès à partir des disques, ventilé par périphérique.
node_disk_reads_completed_total	Nombre total de lectures effectuées avec succès pour les disques, ventilé par périphérique.
node_disk_writes_completed_total	Nombre total d'écritures effectuées avec succès sur les disques, ventilé par périphérique.
node_disk_written_bytes_total	Nombre total d'octets écrits avec succès sur les disques, ventilé par périphérique.

Métrique	Description/Objectif
node_filesystem_avail_bytes	Espace disponible sur le système de fichiers en octets pour les utilisateurs non root, ventilé par périphérique et point de montage.
node_filesystem_size_bytes	Taille totale du système de fichiers en octets, ventilée par périphérique et point de montage.
node_filesystem_free_bytes	Espace libre sur le système de fichiers en octets, ventilé par périphérique et point de montage.
node_filesystem_files	Nombre total de nœuds de fichiers (inodes) sur le système de fichiers, ventilé par périphérique et point de montage.
node_filesystem_files_free	Nombre de nœuds de fichiers libres (inodes) sur le système de fichiers, ventilés par périphérique et point de montage.
node_filesystem_readonly	Indique si le système de fichiers est monté en lecture seule (1 = lecture seule, 0 = lecture-écriture).
node_filesystem_device_error	Indique si une erreur s'est produite lors de l'obtention des statistiques du système de fichiers (1 = erreur, 0 = succès).

Collectez des statistiques sur les groupes de consommateurs Amazon MSK

Dans le cas d'un exportateur Kafka géré, un collecteur Amazon Managed Service for Prometheus collecte des statistiques sur les groupes de consommateurs à partir de votre cluster Amazon MSK, telles que le décalage des consommateurs, les compensations validées et le nombre de membres de chaque groupe de consommateurs. Amazon Managed Service for Prometheus gère l'exportateur pour vous. Vous n'avez donc pas à installer ou à exploiter vous-même un exportateur. Votre destination peut être un service géré par Amazon pour Prometheus Workspace ou Amazon

CloudWatch. Ces métriques complètent les métriques JMX et Node Exporter que le collecteur extrait d'Open Monitoring.

Le scraper se connecte à votre cluster avec AWS Authentification de gestion des identités et des accès (IAM). Vous accordez au scraper l'accès au cluster à l'aide d'une politique de cluster Amazon MSK. Pour collecter des statistiques sur les groupes de consommateurs, créez une configuration de scrape, créez un scraper, puis accordez au scraper l'accès à votre cluster, comme décrit dans les sections suivantes.

Pour plus d'informations sur les mesures, voir Consumer-group indicateurs. Pour envoyer les statistiques à Amazon CloudWatch, voir Collectez des statistiques sur les groupes de consommateurs Amazon MSK dans CloudWatch.

Conditions préalables

Quelques conditions préalables sont requises pour collecter des statistiques sur les groupes de consommateurs :

- Un cluster Amazon MSK Provisioned actif avec Standard Brokers et Apache Kafka version 3.6.0 ou ultérieure, sur le même compte et AWS La région comme grattoir.
- Le contrôle d'accès IAM a été activé pour le cluster. Vous pouvez maintenir l'authentification du client TLS activée en même temps, afin que les clients qui utilisent des certificats puissent continuer à se connecter.
- Au moins deux sous-réseaux dans différentes zones de disponibilité de l'Amazon VPC du cluster, avec le DNS activé.
- Groupes de sécurité qui permettent au scraper d'atteindre les points de terminaison du broker IAM privé du cluster sur le port 9098. Ajoutez une règle entrante au groupe de sécurité du cluster qui autorise le trafic TCP sur le port 9098 à partir du groupe de sécurité que vous fournissez au scraper.
- Un service géré par Amazon pour l'espace de travail Prometheus dans la région du scraper, et un AWS CLI version qui prend en charge `kafkaConfiguration` dans `create-scraper` et `update-scraper`.
- Autorisation de créer ou de mettre à jour un scraper et de valider le cluster : `kafka:DescribeClusterV2`, `kafka:GetBootstrapBrokers`, `kafka-cluster:Connect`, et `kafka-cluster:DescribeTopic`. Pour ajouter la politique de cluster, vous devez également `kafka:GetClusterPolicy` et `kafka:PutClusterPolicy`. Pour obtenir les autorisations nécessaires pour créer le rôle lié à un service du scraper, voir Service-linked autorisations de rôle.

Le scraper utilise son rôle lié à un service pour se connecter au cluster. Vous ne donnez donc aucun rôle à l'exportateur. Par défaut, le rôle ne dispose d'aucune autorisation pour votre cluster. Vous les accordez Autoriser le scraper à accéder au cluster.

L'exportateur Kafka géré ne prend pas en charge les clusters suivants :

- Clusters sans serveur Amazon MSK. Les clusters sans serveur n'acceptent pas les politiques de cluster avec `kafka-cluster` actions. Utilisez le les mesures qu'Amazon MSK envoie à CloudWatch au lieu de cela.
- Clusters pour lesquels l'authentification IAM n'est pas activée.
- Clusters appartenant à un compte ou à une région différent de celui du scraper.

Les ports Open Monitoring et Broker 11001 et 11002 ne sont nécessaires que si vous collectez également des métriques JMX et Node Exporter.

Création d'une configuration de scrape

Vous fournissez une configuration de scrape lorsque vous créez le scraper. Le collecteur géré se connecte au cluster que vous spécifiez et collecte ses métriques automatiquement. Vous ne spécifiez donc pas de cibles de scrape dans la configuration. La configuration doit inclure une section `scrape_configs` avec une tâche dont `job_name` est exactement `kafka-exporter`. Une configuration qui ne contient pas de tâche nommée `kafka-exporter` est rejeté. Le collecteur définit l'intervalle de raclage pour cette tâche.

Utilisez la configuration scrape pour ajouter des étiquettes et filtrer les métriques collectées. L'exemple suivant permet de conserver le décalage des consommateurs, les compensations validées et l'appartenance à un groupe. Il ajoute également un `clusterArn` label afin que vous puissiez distinguer les clusters lorsque vous interrogez. Remplacez l'ARN du cluster par l'ARN de votre cluster et enregistrez la configuration sous `msk-consumer.yaml`. Utilisez le même ARN de cluster lorsque vous créez le scraper.

```
global:
  scrape_interval: 30s
scrape_configs:
  - job_name: kafka-exporter
    relabel_configs:
      - target_label: clusterArn
        replacement: 'arn:aws:kafka:us-west-2:111122223333:cluster/my-cluster/cluster-
id'
```

```
    metric_relabel_configs:
      - source_labels: [__name__]
        regex: kafka_consumergroup_(current_offset|members|lag)
        action: keep
```

L'exportateur peut également envoyer d'autres mesures, telles que les décalages de sujets et de partitions et le nombre de partitions. Amazon MSK fournit déjà certaines de ces informations via Open Monitoring et le les mesures qu'Amazon MSK envoie à CloudWatch. Pour collecter davantage de statistiques sur les exportateurs, ajoutez-les au `regex`, par exemple `kafka_consumergroup_(current_offset|members|lag)|kafka_topic_partitions`. Chaque métrique que vous ajoutez augmente le nombre de séries que vous ingérez.

Pour ajouter l'exportateur à un scraper existant, ajoutez le `kafka-exporter` passez à la configuration de raclage complète du grattoir et conservez ses autres tâches. Mettez ensuite à jour le grattoir comme décrit dans Mettre à jour ou supprimer l'exportateur. Vous pouvez renommer les métriques dans cette tâche, mais vous ne pouvez pas utiliser de règles de réétiquetage pour modifier la façon dont le collecteur se connecte à l'exportateur.

Créer un scraper

Pour indiquer au scraper le cluster à partir duquel effectuer la collecte, vous spécifiez le cluster dans le `exporters` champ de la demande. Pour un cluster Amazon MSK, fournissez un `kafkaConfiguration` avec le `clusterArn` du cluster. Vous pouvez également définir un `topicFilter` et un `consumerGroupFilter` pour sélectionner des sujets et des groupes de consommateurs par nom. Vous fournissez le réseau (sous-réseaux et groupe de sécurité) séparément dans `source` champ.

Utilisez le `create-scraper` commande pour créer un grattoir à l'aide de AWS Command Line Interface. L'exemple suivant sélectionne les sujets qui commencent par `orders` et des groupes de consommateurs qui commencent par `payments`. Remplacez le *`example`* le contenu, y compris les deux filtres, avec vos propres valeurs de cluster, de réseau, d'espace de travail et de filtre. Pour sélectionner tous les sujets ou tous les groupes de consommateurs, omettez le filtre.

> **Note**
>
> Configurez le groupe de sécurité et les sous-réseaux pour qu'ils correspondent à l'Amazon VPC de votre cluster Amazon MSK. Incluez au moins deux sous-réseaux répartis dans deux zones de disponibilité.

```
aws amp create-scraper \
  --region us-west-2 \
  --alias msk-consumer-metrics \
  --source '{"vpcConfiguration":{"subnetIds":["subnet-subnet-id-1","subnet-subnet-
id-2"],"securityGroupIds":["sg-security-group-id"]}}' \
  --exporters '[{"kafkaConfiguration":{"clusterArn":"arn:aws:kafka:us-
west-2:111122223333:cluster/my-cluster/cluster-id","topicFilter":"^orders.*
$","consumerGroupFilter":"^payments.*$"}}]' \
  --cli-binary-format base64 \
  --scrape-configuration "configurationBlob=$(base64 < msk-consumer.yaml | tr -d '\n')"
 \
  --destination '{"ampConfiguration":{"workspaceArn":"arn:aws:aps:us-
west-2:111122223333:workspace/workspace-id"}}'
```

Les deux filtres sont facultatifs. Si vous omettez un filtre, le scraper sélectionne tous les sujets ou tous les groupes de consommateurs. Chaque filtre doit être une expression régulière commençant par ^ et se termine par $, avec une longueur maximale de 256 caractères. Un grattoir peut avoir un seul exportateur.

Le `topicFilter` sélectionne les sujets pour lesquels l'exportateur lit les métriques thématiques. Consumer-group les mesures peuvent également inclure d'autres sujets consommés par un groupe de consommateurs sélectionné. Pour contrôler les indicateurs qui atteignent votre destination, utilisez `metric_relabel_configs` dans votre configuration de scrape.

Enregistrez l'identifiant du scraper. Vérifiez son état à l'aide de la commande suivante et continuez lorsque le statut est `ACTIVE`:

```
aws amp describe-scraper \
  --region us-west-2 \
  --scraper-id scraper-id \
  --query 'scraper.status.statusCode' --output text
```

Le scraper ne collecte pas les métriques Kafka tant que vous ne lui accordez pas l'accès au cluster.

Autoriser le scraper à accéder au cluster

Le scraper a besoin d'une autorisation pour se connecter à votre cluster et pour lire les métadonnées relatives à son sujet et à son groupe de consommateurs. Ajoutez une politique de cluster Amazon MSK qui autorise le rôle lié aux services du scraper à effectuer les actions suivantes :

- `kafka-cluster:Connect` sur le cluster

- `kafka-cluster:DescribeTopic` sur ses sujets
- `kafka-cluster:DescribeGroup` sur ses groupes de consommateurs

Sans cette politique, le scraper est `ACTIVE` mais ne collecte pas les métriques de Kafka. Une fois que vous avez ajouté la politique, le scraper commence à collecter automatiquement et les premières mesures peuvent mettre quelques minutes à apparaître. Il n'est pas nécessaire de mettre à jour le grattoir.

Accordez au scraper un rôle lié aux services, même si le scraper utilise `roleConfiguration` pour envoyer des statistiques à l'espace de travail d'un autre compte. N'accordez pas le rôle source ou le rôle cible.

Accordez un grattoir

Obtenez l'ARN du rôle du scraper avec ce qui suit AWS CLI commande :

```
aws amp describe-scraper \
  --region us-west-2 \
  --scraper-id scraper-id \
  --query scraper.roleArn --output text
```

Chaque grattoir a son propre rôle, `AWSServiceRoleForAmazonPrometheusScraper_`*`suffix`*. Si vous recréez un scraper, attribuez le nouveau rôle.

Enregistrez la politique suivante sous `policy.json`. Remplacez l'ARN du rôle, la région, l'ID de compte, le nom du cluster et l'ID du cluster.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": { "AWS": "scraper-role-arn" },
      "Action": "kafka-cluster:Connect",
      "Resource": "arn:aws:kafka:us-west-2:111122223333:cluster/my-cluster/cluster-id"
    },
    {
      "Effect": "Allow",
      "Principal": { "AWS": "scraper-role-arn" },
      "Action": "kafka-cluster:DescribeTopic",
      "Resource": "arn:aws:kafka:us-west-2:111122223333:topic/my-cluster/cluster-id/*"
```

```
    },
    {
      "Effect": "Allow",
      "Principal": { "AWS": "scraper-role-arn" },
      "Action": "kafka-cluster:DescribeGroup",
      "Resource": "arn:aws:kafka:us-west-2:111122223333:group/my-cluster/cluster-id/*"
    }
  ]
}
```

Pour attribuer un autre scraper, ajoutez à nouveau les trois instructions avec l'ARN du rôle de ce scraper. Pour accorder moins de sujets ou de groupes de consommateurs, modifiez la politique comme décrit dans Limitez les sujets et les groupes de consommateurs avant de l'appliquer.

Appliquer la politique

> **Important**
>
> N'utilisez pas le principal racine d'un compte, tel que `arn:aws:iam::111122223333:root`. Amazon MSK l'accepte, mais n'accorde rien au scraper. N'utilisez pas de caractère générique dans l'ARN d'un rôle en tant que principal. Amazon MSK le rejette avec l'erreur `The cluster policy is not valid`. Pour attribuer chaque scraper du compte, consultez Facultatif : accordez chaque scraper du compte.

Pour lire et modifier la politique de cluster, vous devez `kafka:GetClusterPolicy` et `kafka:PutClusterPolicy`.

Pour ajouter la politique de cluster

1. Vérifiez si le cluster dispose déjà d'une politique :

   ```
   aws kafka get-cluster-policy \
     --region us-west-2 \
     --cluster-arn arn:aws:kafka:us-west-2:111122223333:cluster/my-cluster/cluster-id
   ```

   UNE `NotFoundException` Une erreur signifie que le cluster n'a aucune politique.
2. Si le cluster n'a aucune politique, ajoutez la vôtre :

   ```
   aws kafka put-cluster-policy \
   ```

```
  --region us-west-2 \
  --cluster-arn arn:aws:kafka:us-west-2:111122223333:cluster/my-cluster/cluster-id
 \
  --policy file://policy.json
```

3. Si le cluster dispose déjà d'une politique, fusionnez les instructions. `put-cluster-policy` remplace l'ensemble du document de politique. Le `Policy` valeur de `get-cluster-policy` est la politique existante sous forme de chaîne JSON. Ajoutez ses déclarations au `Statement` liste dans `policy.json`, de sorte que le fichier contienne à la fois les instructions existantes et les nouvelles. Passez ensuite le `CurrentVersion` valeur de `get-cluster-policy`:

```
aws kafka put-cluster-policy \
  --region us-west-2 \
  --cluster-arn arn:aws:kafka:us-west-2:111122223333:cluster/my-cluster/cluster-id
 \
  --current-version current-version \
  --policy file://policy.json
```

Facultatif : accordez chaque scraper du compte

Nous vous recommandons d'autoriser l'accès aux ARN des rôles de scraper individuels. Si vous devez autoriser tous les scrapers actuels et futurs du compte, vous pouvez utiliser la politique suivante à la place. Vous pouvez l'ajouter avant de créer un grattoir, et il continue de fonctionner lorsque vous en remplacez un. Conservez les deux conditions dans chaque déclaration. Enregistrez la politique suivante sous `policy.json`. Remplacez la région, l'ID de compte, le nom du cluster et l'ID du cluster dans chaque déclaration, et appliquez-les comme décrit dans Appliquer la politique.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": { "AWS": "*" },
      "Action": "kafka-cluster:Connect",
      "Resource": "arn:aws:kafka:us-west-2:111122223333:cluster/my-cluster/cluster-id",
      "Condition": {
        "StringEquals": { "aws:PrincipalAccount": "111122223333" },
        "ArnLike": { "aws:PrincipalArn": "arn:aws:iam::111122223333:role/aws-service-
role/scraper.aps.amazonaws.com/AWSServiceRoleForAmazonPrometheusScraper_*" }
      }
    },
```

```
    {
      "Effect": "Allow",
      "Principal": { "AWS": "*" },
      "Action": "kafka-cluster:DescribeTopic",
      "Resource": "arn:aws:kafka:us-west-2:111122223333:topic/my-cluster/cluster-id/*",
      "Condition": {
        "StringEquals": { "aws:PrincipalAccount": "111122223333" },
        "ArnLike": { "aws:PrincipalArn": "arn:aws:iam::111122223333:role/aws-service-
role/scraper.aps.amazonaws.com/AWSServiceRoleForAmazonPrometheusScraper_*" }
      }
    },
    {
      "Effect": "Allow",
      "Principal": { "AWS": "*" },
      "Action": "kafka-cluster:DescribeGroup",
      "Resource": "arn:aws:kafka:us-west-2:111122223333:group/my-cluster/cluster-id/*",
      "Condition": {
        "StringEquals": { "aws:PrincipalAccount": "111122223333" },
        "ArnLike": { "aws:PrincipalArn": "arn:aws:iam::111122223333:role/aws-service-
role/scraper.aps.amazonaws.com/AWSServiceRoleForAmazonPrometheusScraper_*" }
      }
    }
  ]
}
```

Cette politique autorise tous les scrapers actuels et futurs du compte. Le `aws:PrincipalAccount` et `ArnLike` ce sont les conditions qui le limitent. Ne supprimez aucune des deux conditions.

Limitez les sujets et les groupes de consommateurs

Les deux politiques accordent la priorité à tous les sujets et à tous les groupes de consommateurs du cluster. Pour en accorder moins, remplacez la version finale `*` dans les ARN du sujet et du groupe, par exemple `topic/my-cluster/cluster-id/orders-*`. Ces modèles ARN sont des caractères génériques IAM, et non `topicFilter` ou `consumerGroupFilter` expressions régulières. Si la politique accorde moins de sujets ou de groupes que ce que les filtres sélectionnent, le scraper rapporte moins de sujets et de décalages.

Pour restreindre une politique que vous avez déjà appliquée, remplacez vos déclarations précédentes par les déclarations restreintes. Ne les fusionnez pas, car les déclarations précédentes accordent toujours tous les sujets et groupes.

Valider la collecte des métriques

Pour vérifier que le scraper collecte des métriques, exécutez chacune des requêtes ProMQL suivantes séparément sur votre espace de travail. Pour plus d'informations, voir Interrogez vos statistiques Prometheus. Remplacez l'ARN du cluster. Les statistiques apparaissent uniquement pour les groupes de consommateurs qui correspondent à votre filtre de groupes de consommateurs et qui comptent des membres ou des compensations engagées.

```
kafka_consumergroup_members{clusterArn="arn:aws:kafka:us-
west-2:111122223333:cluster/my-cluster/cluster-id"}
```

```
kafka_consumergroup_current_offset{clusterArn="arn:aws:kafka:us-
west-2:111122223333:cluster/my-cluster/cluster-id"}
```

```
kafka_consumergroup_lag{clusterArn="arn:aws:kafka:us-west-2:111122223333:cluster/my-
cluster/cluster-id"}
```

Les premières mesures peuvent mettre quelques minutes à apparaître une fois que vous avez ajouté la politique de cluster. Si aucune statistique n'apparaît par la suite, vérifiez les points suivants. Un résultat vide ne signifie pas que le retard des consommateurs est nul.

- Si le grattoir est `ACTIVE`, mais `up{job="kafka-exporter"}` est `0` et non `kafka_*` les métriques apparaissent, vérifiez la politique du cluster. Une demande de création ou de mise à jour réussie ne signifie pas que le scraper peut accéder au cluster. Courez `aws kafka get-cluster-policy`, et confirmez que la politique accorde `kafka-cluster:Connect`, `kafka-cluster:DescribeTopic`, et `kafka-cluster:DescribeGroup` à l'ARN du rôle lié aux services actuel du scraper. Si vous avez recréé le scraper, son rôle ARN a changé. Si vous utilisez la politique applicable à l'ensemble du compte, assurez-vous que chaque relevé conserve les deux `StringEquals` sur `aws:PrincipalAccount` et `ArnLike` sur `aws:PrincipalArn`. Vous n'avez pas besoin de mettre à jour le scraper après avoir corrigé la politique.
- Si la création ou la mise à jour échoue en raison d'une erreur d'autorisations, accordez à l'appelant les autorisations de scraper et de cluster dans Prérequis, puis réessayez la commande.
- Si le scraper ne peut pas se connecter aux courtiers, activez l'authentification IAM pour le cluster. Vérifiez le DNS du VPC et autorisez le groupe de sécurité du scraper à atteindre les points de terminaison privés du courtier IAM sur le port 9098.
- Si le grattoir est `ACTIVE` mais les requêtes ne renvoient aucune métrique, vérifiez que `consumerGroupFilter` correspond à un groupe de consommateurs existant. Conservez les

noms des métriques et les `clusterArn`, `consumergroup`, `topic`, et `partition` étiquettes dans la configuration Scrape. Utilisez le même ARN de cluster dans la configuration et la requête. Si la politique de cluster accorde moins de sujets ou de groupes que ce que les filtres sélectionnent, moins de sujets et de décalages apparaissent.

- Si la politique de cluster est correcte mais `up{job="kafka-exporter"}` est toujours `0` sur un groupe comprenant de nombreux sujets et groupes de consommateurs, essayez de restreindre `topicFilter` et `consumerGroupFilter`.

Mettre à jour ou supprimer l'exportateur

Pour modifier les filtres, mettez à jour le scraper avec sa configuration complète. La commande suivante permet de sélectionner un sujet et un groupe de consommateurs. Gardez le `kafka-exporter` travail et `clusterArn` étiquette dans votre configuration de scrape.

```
aws amp update-scraper \
  --region us-west-2 \
  --scraper-id scraper-id \
  --exporters '[{"kafkaConfiguration":{"clusterArn":"arn:aws:kafka:us-
west-2:111122223333:cluster/my-cluster/cluster-id","topicFilter":"^orders
$","consumerGroupFilter":"^payments$"}}]' \
  --cli-binary-format base64 \
  --scrape-configuration "configurationBlob=$(base64 < msk-consumer.yaml | tr -d '\n')"
```

En omettant `exporters` dans une mise à jour conserve l'exportateur actuel. Pour le supprimer, passez une liste vide, `--exporters '[]'`, et envoyez la configuration complète du scrape avec `kafka-exporter` tâche supprimée. Conservez tous les autres emplois dont vous avez encore besoin.

Ce qui suit AWS CLI La commande supprime le scraper et arrête toute sa collecte de métriques. Il ne supprime pas le cluster Amazon MSK ni l'espace de travail de destination.

```
aws amp delete-scraper \
  --region us-west-2 \
  --scraper-id scraper-id
```

Supprimer l'accès du scraper au cluster

La suppression de la politique de cluster n'arrête pas immédiatement la collecte. Pour arrêter la collecte, supprimez l'exportateur ou supprimez d'abord le racleur. Supprimez ensuite les déclarations de politique.

Pour supprimer l'accès au grattoir

1. Notez le rôle du scraper (ARN). Supprimez ensuite l'exportateur ou supprimez le racleur, comme décrit plus haut dans cette section.
2. Exécutez `aws amp describe-scraper`. Vérifiez que l'état du grattoir est `ACTIVE` et que le grattoir n'a pas d'exportateur, ou que le grattoir n'existe plus. D'autres scrapers peuvent continuer à envoyer des métriques pour le même cluster.
3. Exécutez `aws kafka get-cluster-policy`. Supprimez uniquement les instructions que vous avez ajoutées pour le grattoir. Conservez les relevés qui donnent accès à d'autres clients. Si vous avez utilisé la politique à l'échelle du compte et que d'autres scrapers ont toujours besoin d'un accès, remplacez ses instructions par des instructions qui nomment le rôle ARN de chaque scraper qui conserve l'accès.
4. Courez `aws kafka put-cluster-policy` avec les déclarations restantes et le `CurrentVersion` valeur, comme dans Appliquer la politique. S'il ne reste aucune instruction, supprimez plutôt la politique avec `aws kafka delete-cluster-policy`. Cette commande nécessite `kafka:DeleteClusterPolicy` autorisation.

Consumer-group indicateurs

Lorsque vous collectez des statistiques sur les groupes de consommateurs à l'aide de l'exemple de configuration dans Collectez des statistiques sur les groupes de consommateurs Amazon MSK, le collecteur Amazon Managed Service for Prometheus envoie les métriques suivantes. Chaque métrique possède `clusterArn` label à partir de la configuration de scrape, afin que vous puissiez distinguer les clusters.

Métrique	Description/Objectif
kafka_consumergroup_lag	Différence entre la compensation la plus récente d'une partition et la compensation engagée par le groupe de consommateurs. Il s'agit approximativement du nombre de

Métrique	Description/Objectif
	messages que le groupe n'a pas encore validés. Étiquettes : `consumergroup` , `topic`, `partition` .
kafka_consumergroup_current_offset	Dernière compensation que le groupe de consommateurs s'est engagée à effectuer une partition. Étiquettes : `consumergroup` , `topic`, `partition` .
membres_du groupe de consommateurs kafka_	Nombre de membres du groupe de consommateurs. Libellé : `consumergroup` .

Une valeur de `-1` pour le décalage ou la compensation validée, cela signifie que le groupe de consommateurs n'a pas encore engagé de compensation pour cette partition. Oubliez ces valeurs lorsque vous additionnez le décalage. Par exemple, la requête suivante indique le décalage total de chaque groupe de consommateurs pour chaque sujet :

```
sum by (clusterArn, consumergroup, topic) (kafka_consumergroup_lag >= 0)
```

Ce total n'inclut pas les partitions pour lesquelles le groupe de consommateurs n'a pas engagé de compensation.

L'exportateur calcule le décalage à partir des décalages qu'il lit à partir de votre cluster. Ses valeurs peuvent différer de indicateurs de retard des consommateurs fournis par Amazon MSK, car chacun lit les décalages à des moments différents.

Limitations

L'intégration actuelle d'Amazon MSK à Amazon Managed Service pour Prometheus présente les limites suivantes :

- Uniquement pris en charge pour les clusters Amazon MSK Provisioned (non disponible pour Amazon MSK Serverless)
- Non pris en charge pour les clusters Amazon MSK dont l'accès public est activé en combinaison avec le mode de métadonnées Kraft

- Supporte actuellement un mappage 1:1 entre les clusters Amazon MSK et Amazon Managed Service for Prometheus collectors/workspaces

Configurer des collecteurs Prometheus gérés pour Amazon OpenSearch Service

Le service géré Amazon pour Prometheus, collecteur géré pour Amazon OpenSearch Le service supprime automatiquement Prometheus-compatible les statistiques de votre OpenSearch Domaine de service et les redirige vers votre destination. Amazon Managed Service for Prometheus gère le collecteur pour vous, vous offrant ainsi l'évolutivité, la sécurité et la fiabilité dont vous avez besoin, sans avoir à gérer vous-même des instances, des agents ou des scrapers. Votre destination peut être un service géré par Amazon pour Prometheus Workspace ou Amazon CloudWatch.

Vous pouvez également créer un scraper qui s'intègre à Amazon Elastic Kubernetes Service ou à Amazon Managed Streaming pour Apache Kafka. Pour plus d'informations, voir Intégrez Amazon EKS et Intégrez Amazon MSK.

Créer un scraper

Cette procédure suppose que vous connaissez Amazon OpenSearch Administration des domaines de service et concepts de mise en réseau Amazon Virtual Private Cloud.

Il existe quelques prérequis pour créer un grattoir pour un OpenSearch Domaine de service :

- Une Amazon OpenSearch Domaine de service avec accès à Amazon Virtual Private Cloud. Support pour les collecteurs gérés uniquement OpenSearch Domaines de service dotés d'un accès VPC. Les domaines avec accès public ne sont pas pris en charge.
- Groupes de sécurité qui permettent au collecteur géré d'accéder à votre OpenSearch Point de terminaison du domaine de service via HTTPS (port 443). Ajoutez au groupe de sécurité du domaine une règle entrante autorisant le trafic HTTPS provenant du groupe de sécurité que vous fournissez pour le collecteur.

Pour indiquer au grattoir quel OpenSearch Domaine de service auprès duquel effectuer la collecte, vous spécifiez le domaine dans le `exporters` champ de la demande. Le `exporters` le champ prend une liste des configurations de l'exportateur ; pour un OpenSearch Domaine de service, fournissez un `openSearchConfiguration` avec le `domainArn` du domaine. Vous fournissez le

réseau (sous-réseaux et groupe de sécurité) séparément dans `source` champ, comme illustré dans les exemples suivants.

> **Note**
>
> Le scraper utilise le point de terminaison VPC du domaine tel qu'il a été résolu lorsque vous créez le scraper. Si vous recréez le domaine, il obtient un nouveau point de terminaison. Vous devez donc créer un nouveau scraper ou mettre à jour le scraper existant avec UpdateScraper à collecter auprès de celui-ci.

Vous fournissez une configuration de scrape lorsque vous créez le scraper. Le collecteur géré se connecte au domaine que vous spécifiez et collecte automatiquement ses métriques ; vous ne spécifiez donc pas de cibles d'extraction dans la configuration. La configuration doit inclure une section `scrape_configs` avec une tâche dont `job_name` est exactement `opensearch-exporter`. Une configuration qui ne contient pas de tâche nommée `opensearch-exporter` est rejeté. Utilisez la configuration de scrape pour définir l'intervalle de scrape et, éventuellement, pour filtrer ou renommer les métriques collectées. Voici un exemple de configuration de scrape :

```
global:
  external_labels:
    domain_name: my-opensearch-domain

scrape_configs:
  - job_name: opensearch-exporter
    scrape_interval: 60s
```

Pour plus d'informations sur les options de configuration de scrape prises en charge, consultez Configuration du scraper.

To create a scraper using the AWS API

Utilisez le `CreateScraper` Fonctionnement de l'API pour créer un scraper avec AWS API. L'exemple suivant crée un scraper dans la région USA Est (Virginie du Nord) qui collecte des métriques à partir d'un OpenSearch Domaine de service et les envoie à un espace de travail Amazon Managed Service pour Prometheus. Remplacez le *`example`* du contenu avec vos propres informations de domaine, de réseau et d'espace de travail, et fournissez la configuration de votre scraper.

> **Note**
>
> Configurez le groupe de sécurité et les sous-réseaux en fonction de l'Amazon VPC de votre OpenSearch Domaine de service. Incluez au moins deux sous-réseaux dans deux zones de disponibilité.

```
POST /scrapers HTTP/1.1

{
    "alias": "myScraper",
    "source": {
        "vpcConfiguration": {
            "securityGroupIds": ["sg-security-group-id"],
            "subnetIds": ["subnet-subnet-id-1", "subnet-subnet-id-2"]
        }
    },
    "exporters": [
        {
            "openSearchConfiguration": {
                "domainArn": "arn:aws:es:us-east-1:123456789012:domain/my-
opensearch-domain"
            }
        }
    ],
    "destination": {
        "ampConfiguration": {
            "workspaceArn": "arn:aws:aps:us-east-1:123456789012:workspace/ws-
workspace-id"
        }
    },
    "scrapeConfiguration": {
        "configurationBlob": "base64-encoded-blob"
    }
}
```

Pour envoyer les statistiques à Amazon CloudWatch au lieu d'un espace de travail Amazon Managed Service pour Prometheus, remplacez `destination` avec un CloudWatch configuration :

```
    "destination": {
```

```
        "cloudWatchConfiguration": {
            "datasetArn": "arn:aws:cloudwatch:us-east-1:123456789012:dataset/
default"
        }
    }
```

Le `scrapeConfiguration` Le paramètre nécessite un fichier YAML de configuration Prometheus codé en base64. Exécutez l'une des commandes suivantes pour convertir le fichier YAML en base64. Vous pouvez également utiliser n'importe quel convertisseur base64 en ligne pour convertir le fichier.

Example Linux/macOS

```
base64 -w0 scraper-config.yaml
```

Example Fenêtres PowerShell

```
[Convert]::ToBase64String([System.IO.File]::ReadAllBytes("scraper-config.yaml"))
```

To create a scraper using the AWS CLI

Utilisez le `create-scraper` commande pour créer un grattoir à l'aide de AWS Command Line Interface. L'exemple suivant crée un scraper dans la région de l'Est des États-Unis (Virginie du Nord). Remplacez le *example* du contenu avec vos propres informations de domaine, de réseau et d'espace de travail, et fournissez la configuration de votre scraper.

> **Note**
>
> Configurez le groupe de sécurité et les sous-réseaux en fonction de l'Amazon VPC de votre OpenSearch Domaine de service. Incluez au moins deux sous-réseaux dans deux zones de disponibilité.

```
aws amp create-scraper \
 --source '{"vpcConfiguration":{"securityGroupIds":["sg-security-group-
id"],"subnetIds":["subnet-subnet-id-1","subnet-subnet-id-2"]}}' \
 --exporters '[{"openSearchConfiguration":{"domainArn":"arn:aws:es:us-
east-1:123456789012:domain/my-opensearch-domain"}}]' \
 --scrape-configuration configurationBlob=base64-encoded-blob \
```

```
 --destination '{"ampConfiguration":{"workspaceArn":"arn:aws:aps:us-
east-1:123456789012:workspace/ws-workspace-id"}}'
```

- Vous trouverez ci-dessous une liste complète des opérations de grattage que vous pouvez utiliser avec AWS API :

  Créez un scraper avec l'opération d'API CreateScraper.
- Répertoriez vos scrapers existants avec l'opération d'API ListScrapers.
- Mettez à jour l'alias, la configuration ou la destination d'un scraper avec UpdateScraper Fonctionnement de l'API.
- Supprimez un scraper avec l'opération d'API DeleteScraper.
- Obtenez plus de détails sur un scraper avec l'opération d'API DescribeScraper.

Cross-account installation

Pour créer un scraper dans une configuration multi-comptes, lorsque le OpenSearch Le domaine de service à partir duquel vous souhaitez collecter des métriques se trouve dans un compte différent de celui du collecteur Amazon Managed Service for Prometheus. Suivez la procédure suivante.

Par exemple, vous avez deux comptes : un compte source `account_id_source` où le OpenSearch Le domaine de service est localisé et un compte cible `account_id_target` où se trouve l'espace de travail Amazon Managed Service for Prometheus.

> **Note**
>
> Les politiques de confiance décrites dans les étapes suivantes font référence au scraper par son ARN, qui comprend un *`scraper-id`* qui n'existe pas tant que vous n'avez pas créé le grattoir. Pour éviter ce problème d'ordre, créez d'abord les rôles avec un caractère générique (`scraper/*`) à la place de l'ARN de scraper spécifique, créez le scraper, puis mettez à jour les deux politiques de confiance pour remplacer le caractère générique par l'ARN de scraper réel qui `CreateScraper` retours.

Pour créer un scraper dans une configuration multi-comptes

1. Dans le compte source, créez un rôle `arn:aws:iam::`*`111122223333`*`:role/Source` et ajoutez la politique de confiance suivante.

```
{
    "Effect": "Allow",
    "Principal": {
    "Service": [
        "scraper.aps.amazonaws.com"
     ]
    },
    "Action": "sts:AssumeRole",
    "Condition": {
        "ArnEquals": {
            "aws:SourceArn": "arn:aws:aps:aws-region:111122223333:scraper/scraper-id"
        },
        "StringEquals": {
            "AWS:SourceAccount": "111122223333"
        }
    }
}
```

2. Dans le compte cible, créez un rôle `arn:aws:iam::444455556666:role/Target` et ajoutez la politique de confiance suivante, qui permet au rôle source de l'assumer.

```
{
  "Effect": "Allow",
  "Principal": {
     "AWS": "arn:aws:iam::111122223333:role/Source"
  },
  "Action": "sts:AssumeRole",
  "Condition": {
     "StringEquals": {
        "sts:ExternalId": "arn:aws:aps:aws-region:111122223333:scraper/scraper-id"
      }
  }
}
```

Associez une politique d'autorisations au rôle cible qui lui permet d'écrire vers votre destination. Si votre destination est un espace de travail Amazon Managed Service pour Prometheus, joignez AmazonPrometheusRemoteWriteAccess (qui accorde `aps:RemoteWrite`). Si votre destination est Amazon CloudWatch, joignez une politique qui accorde `cloudwatch:PutMetricData` sur l'ensemble de données.

3. Créez un grattoir à l'aide du `--role-configuration` option.

```
aws amp create-scraper \
 --source '{"vpcConfiguration":{"securityGroupIds":["sg-security-group-
id"],"subnetIds":["subnet-subnet-id-1","subnet-subnet-id-2"]}}' \
 --exporters '[{"openSearchConfiguration":{"domainArn":"arn:aws:es:aws-
region:111122223333:domain/my-opensearch-domain"}}]' \
 --scrape-configuration configurationBlob=<base64-encoded-blob> \
 --destination '{"ampConfiguration":{"workspaceArn":"arn:aws:aps:aws-
region:444455556666:workspace/ws-workspace-id"}}' \
 --role-configuration '{"sourceRoleArn":"arn:aws:iam::111122223333:role/Source",
 "targetRoleArn":"arn:aws:iam::444455556666:role/Target"}'
```

4. Validez la création du scraper.

```
aws amp list-scrapers
```

Recherche et suppression des scrapers

Vous pouvez utiliser le AWS API ou AWS CLI pour répertorier les scrapers de votre compte ou pour les supprimer.

> **Note**
>
> Assurez-vous que vous utilisez la dernière version du AWS CLI ou SDK. La dernière version vous fournit les dernières fonctionnalités et fonctionnalités, ainsi que des mises à jour de sécurité. Vous pouvez également utiliser AWS CloudShell, qui fournit automatiquement une expérience de ligne de commande toujours à jour.

Pour afficher tous les scrapers de votre compte, utilisez l'opératon d'API ListScrapers. Alternativement, avec AWS CLI, appelez :

```
aws amp list-scrapers
```

Pour supprimer un scraper, recherchez le `scraperId` du scraper que vous souhaitez supprimer à l'aide de l'opération `ListScrapers`, puis utilisez l'opération DeleteScraper pour le supprimer. Alternativement, avec AWS CLI, appelez :

```
aws amp delete-scraper --scraper-id scraperId
```

Statistiques collectées auprès d'Amazon OpenSearch Service

Lorsque vous intégrez Amazon OpenSearch Service, le collecteur Amazon Managed Service for Prometheus, supprime automatiquement Prometheus-compatible des indicateurs qui décrivent l'état et les performances de votre domaine. Le collecteur émet plusieurs centaines de métriques ; les tableaux suivants répertorient les métriques représentatives de chaque catégorie. Le `opensearch_indices_` les métriques des tableaux suivants sont agrégées sur l'ensemble du domaine, et nombre d'entre elles ont des variantes principales uniquement et totales (suffixé) `_primary` ou `_total`). Le collecteur émet également les mêmes statistiques par index, avec le préfixe `opensearch_index_stats_`, et des métriques par fragment préfixées `opensearch_indices_shards_`.

Métriques du cluster

Métrique	Description/Objectif
opensearch_cluster_health_status	État de santé du cluster (vert, jaune ou rouge).
opensearch_cluster_health_number_of_nodes	Nombre de nœuds dans le cluster .
opensearch_cluster_health_number_of_ data_nodes	Nombre de nœuds de données dans le cluster.
opensearch_cluster_health_active_pri mary_shards	Nombre de fragments principaux actifs.
opensearch_cluster_health_active_shards	Nombre total de fragments actifs, y compris les fragments principaux et les répliques.
opensearch_cluster_health_relocating_shards	Nombre de partitions qui sont déplacées entre les nœuds.
opensearch_cluster_health_initializing_shards	Nombre de partitions en cours d'initialisation.
opensearch_cluster_health_unassigned_shards	Nombre de partitions qui ne sont pas attribuées à un nœud.

Métrique	Description/Objectif
opensearch_cluster_health_delayed_unassigned_shards	Nombre de partitions non attribuées dont l'affectation est retardée.
opensearch_cluster_health_number_of_pending_tasks	Nombre de tâches au niveau du cluster mises en file d'attente et en attente d'exécution.
opensearch_cluster_health_number_of_in_flight_fetch	Nombre de demandes d'extraction de partitions en cours.
opensearch_cluster_health_task_max_waiting_in_queue_millis	Durée d'attente la plus longue d'une tâche dans la file d'attente, en millisecondes.

Métriques des nœuds

Métrique	Description/Objectif
opensearch_os_cpu_percent	Pourcentage de CPU utilisé par le système d'exploitation sur le nœud.
opensearch_os_load1	Charge moyenne du système d'exploitation au cours de la dernière minute. Le collecteur émet également `opensearch_os_load5` et `opensearch_os_load15` .
opensearch_os_mem_used_bytes	Quantité de mémoire physique utilisée, en octets. Le collecteur émet également `opensearch_os_mem_free_bytes` , `opensearch_os_mem_actual_used_bytes` , et `opensearch_os_mem_actual_free_bytes` .
opensearch_jvm_memory_used_bytes	Quantité de mémoire JVM utilisée, en octets, par zone mémoire (tas et non-tas). Les métriques associées incluent `opensearc`

Métrique	Description/Objectif
	`h_jvm_memory_committed_bytes` et `opensearch_jvm_memory_max_bytes` .
opensearch_jvm_gc_collection_seconds_count	Nombre total d'événements de collecte des déchets de la JVM. À utiliser avec `opensearch_jvm_gc_collection_seconds_sum` , le temps total consacré à la collecte des ordures.
opensearch_jvm_uptime_seconds	Temps de disponibilité de la JVM, en secondes.
opensearch_thread_pool_active_count	Nombre de threads actifs dans chaque pool de threads. Les mesures connexes incluent `opensearch_thread_pool_queue_count` , `opensearch_thread_pool_rejected_count` , et `opensearch_thread_pool_completed_count` .
opensearch_filesystem_data_available_bytes	Quantité d'espace disque disponible pour le nœud, en octets. Les métriques associées incluent `opensearch_filesystem_data_free_bytes` et `opensearch_filesystem_data_size_bytes` .
opensearch_filesystem_io_stats_device_read_operations_count	Nombre d'opérations de lecture sur disque. Le collecteur émet également le nombre total d'opérations d'écriture et le nombre total d'opérations ainsi que les tailles de lecture et d'écriture (par exemple, `opensearch_filesystem_io_stats_device_read_size_kilobytes_sum`).
opensearch_process_cpu_percent	Pourcentage de CPU utilisé par OpenSearch processus.

Métrique	Description/Objectif
opensearch_process_mem_resident_size_bytes	Taille de la mémoire résidente du OpenSearch processus, en octets.
opensearch_process_open_files_count	Nombre de descripteurs de fichiers ouverts par OpenSearch processus.
opensearch_breakers_tripped	Nombre total de fois que chaque disjoncteur s'est déclenché. Les métriques associées incluent `opensearch_breakers_estimated_size_bytes` et `opensearch_breakers_limit_size_bytes` .
opensearch_transport_rx_size_bytes_total	Quantité totale de données reçues sur la couche de transport entre les nœuds, en octets. Le collecteur émet également `opensearch_transport_tx_size_bytes_total` et le nombre de paquets.
opensearch_indexing_pressure_current_all_in_bytes	Mémoire actuelle consommée par les demandes d'indexation, en octets, avec `opensearch_indexing_pressure_limit_in_bytes` comme limite configurée.

Métriques d'index

Métrique	Description/Objectif
opensearch_indices_docs	Nombre de documents Les mesures connexes incluent `opensearch_indices_docs_deleted` , `opensearch_indices_docs_primary` , et `opensearch_indices_docs_total` .

Métrique	Description/Objectif
opensearch_indices_store_size_bytes	Taille totale des indices sur le disque, en octets, avec `opensearch_indices_store_size_bytes_primary` et `opensearch_indices_store_size_bytes_total` variantes.
opensearch_indices_indexing_index_total	Nombre total de documents indexés. À utiliser avec `opensearch_indices_indexing_index_time_seconds_total` pour indexer la latence.
opensearch_indices_indexing_is_throttled	Indique si l'indexation est actuellement limitée, avec `opensearch_indices_indexing_throttle_time_seconds_total` pendant la durée totale d'étranglement.
opensearch_indices_search_query_total	Nombre total de requêtes de recherche. À utiliser avec `opensearch_indices_search_query_time_seconds` pour la latence des requêtes.
opensearch_indices_search_fetch_total	Nombre total d'opérations de récupération, avec `opensearch_indices_search_fetch_time_seconds` pour la latence de récupération.
opensearch_indices_get_total	Nombre total d'opérations get, avec `opensearch_indices_get_time_seconds` pour obtenir de la latence.
opensearch_indices_merges_total	Nombre total de fusions de segments réalisées. Les métriques associées incluent `opensearch_indices_merges_current` et `opensearch_indices_merges_total_time_seconds_total` .

Métrique	Description/Objectif
opensearch_indices_refresh_total	Nombre total d'opérations d'actualisation de l'index, avec `opensearch_indices_refresh_time_seconds_total` pour le temps de rafraîchissement.
opensearch_indices_flush_total	Nombre total d'opérations de rinçage de l'index, avec `opensearch_indices_flush_time_seconds` pour le temps de rinçage.
opensearch_indices_segments_count	Nombre de segments. Le collecteur émet également des métriques de mémoire par segment (par exemple, `opensearch_indices_segments_memory_bytes` et `opensearch_indices_segment_terms_memory_total`).
opensearch_indices_translog_operations	Nombre d'opérations dans le journal des transactions, avec `opensearch_indices_translog_size_in_bytes` pour sa taille.
opensearch_indices_fielddata_memory_size_bytes	Mémoire utilisée par le cache de données de champ, en octets, avec `opensearch_indices_fielddata_evictions` pour les expulsions.
opensearch_indices_query_cache_memory_size_bytes	Mémoire utilisée par le cache de requêtes, en octets. Les métriques associées incluent `opensearch_indices_query_cache_count` et `opensearch_indices_query_cache_evictions`.

Métrique	Description/Objectif
opensearch_indices_request_cache_memory_size_bytes	Mémoire utilisée par le cache de requêtes, en octets. Les métriques associées incluent `opensearch_indices_request_cache_count` et `opensearch_indices_request_cache_evictions` .

Limitations

L'Amazonie OpenSearch L'intégration des services avec Amazon Managed Service pour Prometheus présente les limites suivantes :

- Uniquement pris en charge pour OpenSearch Domaines de service qui ont accès à Amazon Virtual Private Cloud. Les domaines avec accès public ne sont pas pris en charge.
- Un scraper collecte des métriques à partir d'un seul OpenSearch Domaine de service. Pour collecter les métriques de plusieurs domaines, créez un extracteur distinct pour chaque domaine.

Quels sont Prometheus-compatible des indicateurs ?

Pour extraire les métriques Prometheus de vos applications et de votre infrastructure afin de les utiliser dans Amazon Managed Service for Prometheus, ils doivent instrumenter et exposer Prometheus-compatible indicateurs à partir de Prometheus-compatible `/metrics` points de terminaison. Vous pouvez mettre en œuvre vos propres métriques, mais ce n'est pas obligatoire. Kubernetes (y compris Amazon EKS) et de nombreuses autres bibliothèques et services mettent directement en œuvre ces métriques.

Lorsque les métriques d'Amazon EKS sont exportées vers un Prometheus-compatible point de terminaison, vous pouvez faire extraire automatiquement ces métriques par le collecteur Amazon Managed Service for Prometheus.

Pour plus d'informations, consultez les rubriques suivantes :

- Pour plus d'informations sur les bibliothèques et services existants qui exportent des métriques sous forme de métriques Prometheus, consultez la section Exportateurs et intégrations dans la documentation Prometheus.

- Pour plus d'informations sur l'exportation Prometheus-compatible des métriques issues de votre propre code, voir Exportateurs d'écriture dans la documentation Prometheus.
- Pour plus d'informations sur la configuration d'un collecteur Amazon Managed Service for Prometheus afin de collecter automatiquement des métriques de vos clusters Amazon EKS, consultez la section Configurer des collecteurs gérés pour Amazon EKS.

Surveillez les collecteurs à l'aide de journaux vendus

Amazon Managed Service for Prometheus Collectors fournit des journaux vendus pour vous aider à surveiller et à résoudre les problèmes liés au processus de collecte des métriques. Ces journaux sont automatiquement envoyés à Amazon CloudWatch Logs et fournissent une visibilité sur les opérations de découverte de services, de collecte de métriques et d'exportation de données. Le collecteur vend des journaux pour trois composants principaux du pipeline de collecte de mesures :

Rubriques

- Journaux de découverte des services
- Journaux de collecte
- Journaux des exportateurs
- Comprendre et utiliser les journaux vendus par des collectionneurs

Journaux de découverte des services

Les journaux de découverte de services fournissent des informations sur le processus de découverte de la cible, notamment :

- Problèmes d'authentification ou d'autorisation lors de l'accès aux ressources de l'API Kubernetes.
- Erreurs de configuration dans les paramètres de découverte des services.

Les exemples suivants illustrent les erreurs d'authentification et d'autorisation courantes que vous pouvez rencontrer lors de la découverte de services :

Cluster Amazon EKS inexistant

Lorsque le cluster Amazon EKS spécifié n'existe pas, le message d'erreur suivant s'affiche :

```
{
  "component": "SERVICE_DISCOVERY",
```

```
  "timestamp": "2025-04-30T17:25:41.946Z",
  "message": {
    "log": "Failed to watch Service - Verify your scraper source exists."
  },
  "scrapeConfigId": "s-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Autorisations non valides pour les services

Lorsque le collecteur ne dispose pas des autorisations de contrôle d'accès basé sur les rôles (RBAC) appropriées pour surveiller les services, le message d'erreur suivant s'affiche :

```
{
  "component": "SERVICE_DISCOVERY",
  "timestamp": "2025-04-30T17:25:41.946Z",
  "message": {
    "log": "Failed to watch Service - Verify your scraper source permissions are
 valid."
  },
  "scrapeConfigId": "s-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Autorisations non valides pour les terminaux

Lorsque le collecteur ne dispose pas des autorisations appropriées de contrôle d'accès basé sur les rôles (RBAC) pour surveiller les points de terminaison, le message d'erreur suivant s'affiche :

```
{
  "component": "SERVICE_DISCOVERY",
  "timestamp": "2025-04-30T17:25:41.946Z",
  "message": {
    "log": "Failed to watch Endpoints - Verify your scraper source permissions are
 valid."
  },
  "scrapeConfigId": "s-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Journaux de collecte

Les journaux du collecteur fournissent des informations sur le processus de scraping métrique, notamment :

- Supprimez les défaillances dues à l'indisponibilité des terminaux.
- Problèmes de connexion lors de la tentative de capture de cibles.
- Délais d'attente pendant les opérations de raclage.
- Erreurs d'état HTTP renvoyées par les cibles Scrape.

Les exemples suivants illustrent les erreurs de collecteur courantes que vous pouvez rencontrer lors du processus de capture des métriques :

Point de terminaison des métriques manquant

Lorsque le `/metrics` point de terminaison n'est pas disponible sur l'instance cible, le message d'erreur suivant s'affiche :

```
{
    "component": "COLLECTOR",
    "message": {
        "log": "Failed to scrape Prometheus endpoint - verify /metrics endpoint is
 available",
        "job": "pod_exporter",
        "targetLabels": "{__name__=\"up\", instance=\10.24.34.0\", job=
\"pod_exporter\"}"
    },
    "timestamp": "1752787969551",
    "scraperId": "s-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Connexion refusée

Lorsque le collecteur ne parvient pas à établir une connexion avec le point de terminaison cible, le message d'erreur suivant s'affiche :

```
{
  "scrapeConfigId": "s-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "timestamp": "2025-04-30T17:25:41.946Z",
  "message": {
    "message": "Scrape failed",
    "scrape_pool": "pod_exporter",
    "target": "http://10.24.34.0:80/metrics",
    "error": "Get \"http://10.24.34.0:80/metrics\": dial tcp 10.24.34.0:80: connect:
 connection refused"
  },
```

```
  "component": "COLLECTOR"
}
```

Journaux des exportateurs

Les journaux des exportateurs fournissent des informations sur le processus d'envoi des métriques collectées à votre espace de travail Amazon Managed Service for Prometheus, notamment :

- Nombre de mesures et de points de données traités.
- Échecs d'exportation dus à des problèmes d'espace de travail.
- Erreurs d'autorisation lors de la tentative d'écriture de métriques.
- Défaillances de dépendance dans le pipeline d'exportation.

L'exemple suivant illustre une erreur d'exportation courante que vous pouvez rencontrer lors du processus d'exportation des métriques :

Espace de travail introuvable

Lorsque l'espace de travail cible pour l'exportation des métriques est introuvable, le message d'erreur suivant s'affiche :

```
{
    "component": "EXPORTER",
    "message": {
        "log": "Failed to export to the target workspace - Verify your scraper
 destination.",
        "samplesDropped": 5
    },
    "timestamp": "1752787969664",
    "scraperId": "s-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Comprendre et utiliser les journaux vendus par des collectionneurs

Structure du journal

Tous les journaux vendus par des collectionneurs suivent une structure cohérente avec les champs suivants :

scrapeConfigId

Identifiant unique de la configuration de scrape qui a généré le journal.

timestamp

Heure à laquelle l'entrée du journal a été générée.

message

Le contenu du message du journal, qui peut inclure des champs structurés supplémentaires.

composant

Le composant qui a généré le journal (SERVICE_DISCOVERY, COLLECTOR ou EXPORTER)

Utilisation des journaux vendus pour le dépannage

Les journaux vendus par le collecteur vous aident à résoudre les problèmes courants liés à la collecte des métriques :

1. Problèmes liés à la découverte de services
   - Vérifiez les journaux SERVICE_DISCOVERY pour détecter les erreurs d'authentification ou d'autorisation.
   - Vérifiez que le collecteur dispose des autorisations nécessaires pour accéder aux ressources Kubernetes.
2. Problèmes liés au scraping métrique
   - Consultez les journaux COLLECTOR pour détecter les défaillances liées au scrapage.
   - Vérifiez que les points de terminaison cibles sont accessibles et renvoient des métriques.
   - Assurez-vous que les règles de pare-feu autorisent le collecteur à se connecter aux points de terminaison cibles.
3. Problèmes d'exportation de données métriques
   - Vérifiez les journaux de l'EXPORTATEUR pour détecter les échecs d'exportation.
   - Vérifiez que l'espace de travail existe et qu'il est correctement configuré.
   - Assurez-vous que le collecteur dispose des autorisations nécessaires pour écrire dans l'espace de travail.

Accès aux journaux vendus par les collecteurs

Les journaux vendus par Collector sont automatiquement envoyés à Amazon CloudWatch Logs. Pour accéder à ces journaux :

1. Ouvrez la CloudWatch console à l'adresse https://console.aws.amazon.com/cloudwatch/.
2. Dans le panneau de navigation, choisissez Groupes de journaux.
3. Recherchez et sélectionnez le groupe de journaux pour votre collecteur :`/aws/prometheus/workspace_id/collector/collector_id`.
4. Parcourez ou recherchez les événements du journal pour trouver des informations pertinentes.

Vous pouvez également utiliser CloudWatch Logs Insights pour interroger et analyser les journaux de vos collecteurs. Par exemple, pour rechercher toutes les erreurs de découverte de services :

```
fields @timestamp, message.message
| filter component = "SERVICE_DISCOVERY" and message.message like /Failed/
| sort @timestamp desc
```

Bonnes pratiques pour la surveillance des collecteurs

Pour surveiller efficacement votre Amazon Managed Service pour les collectionneurs Prometheus :

1. Configurez des CloudWatch alarmes pour les problèmes critiques du collecteur, tels que les défaillances persistantes de raclage ou les erreurs d'exportation. Pour plus d'informations, consultez la section Alarmes du guide de CloudWatch l'utilisateur Amazon.
2. Créez des CloudWatch tableaux de bord pour visualiser les indicateurs de performance des collecteurs ainsi que les données des journaux vendus. Pour plus d'informations, consultez la section Tableaux de bord du guide de l' CloudWatch utilisateur Amazon.
3. Consultez régulièrement les journaux de découverte des services pour vous assurer que les cibles sont correctement découvertes.
4. Surveillez le nombre de cibles abandonnées pour identifier les problèmes de configuration potentiels.
5. Suivez les échecs d'exportation pour vous assurer que les métriques sont correctement envoyées à votre espace de travail.

Collecteurs gérés par le client

Cette section contient des informations sur l'ingestion de données en configurant vos propres collecteurs qui envoient des métriques à Amazon Managed Service for Prometheus à l'aide de l'écriture à distance Prometheus.

Lorsque vous utilisez vos propres collecteurs pour envoyer des métriques à Amazon Managed Service for Prometheus, il vous incombe de sécuriser vos métriques et de vous assurer que le processus d'ingestion répond à vos besoins de disponibilité.

La plupart des collecteurs gérés par le client utilisent l'un des outils suivants :

- AWS Distribution pour OpenTelemetry (ADOT) — ADOT est une distribution open source entièrement prise en charge, sécurisée et prête à la production, OpenTelemetry qui fournit des agents pour collecter des métriques. Vous pouvez utiliser ADOT pour collecter des métriques et les envoyer à votre espace de travail Amazon Managed Service for Prometheus. Pour plus d'informations sur l'ADOT Collector, consultez AWS Distro for. OpenTelemetry
- Agent Prometheus – Vous pouvez configurer votre propre instance du serveur Prometheus open source, exécuté en tant qu'agent, pour collecter des métriques et les transmettre à votre espace de travail Amazon Managed Service for Prometheus.

Les rubriques suivantes décrivent l'utilisation de ces deux outils et incluent des informations générales sur la configuration de vos propres collecteurs.

Rubriques

- Sécurisation de l'ingestion de vos métriques
- Utilisation AWS Distro pour en OpenTelemetry tant que collectionneur
- Utilisation d'une instance Prometheus comme collecteur
- Configurez Amazon Managed Service pour Prometheus afin d'obtenir des données à haute disponibilité

Sécurisation de l'ingestion de vos métriques

Amazon Managed Service for Prometheus vous aide à sécuriser l'ingestion de vos métriques.

Utilisation AWS PrivateLink avec Amazon Managed Service pour Prometheus

Le trafic réseau lié à l'ingestion des métriques dans Amazon Managed Service for Prometheus peut être effectué via un point de terminaison Internet public ou via un point de terminaison VPC via. AWS PrivateLink Lorsque vous utilisez AWS PrivateLink , le trafic réseau provenant de vos VPC est sécurisé au sein du réseau AWS sans passer par l'Internet public. Pour créer un point de terminaison AWS PrivateLink VPC pour Amazon Managed Service for Prometheus, consultez. Utilisation d'Amazon Managed Service for Prometheus avec des points de terminaison de VPC d'interface

Authentification et autorisation

AWS La gestion des identités et des accès (IAM) est un service Web qui vous permet de contrôler en toute sécurité l'accès aux AWS ressources. Vous pouvez utiliser IAM pour contrôler les personnes qui s'authentifient (sont connectées) et sont autorisées (disposent d'autorisations) à utiliser des ressources. Amazon Managed Service for Prometheus s'intègre à IAM pour vous aider à protéger vos données. Lorsque vous configurez Amazon Managed Service for Prometheus, vous devez créer des rôles IAM qui lui permettent d'ingérer des métriques des serveurs Prometheus et qui permettent aux serveurs Grafana d'interroger les métriques stockées dans vos espaces de travail Amazon Managed Service for Prometheus. Pour plus d'informations sur IAM, consultez En quoi consiste IAM ?.

Une autre fonctionnalité AWS de sécurité qui peut vous aider à configurer Amazon Managed Service pour Prometheus est le processus de AWS signature Signature Version 4 (AWS SigV4). Signature Version 4 est le processus permettant d'ajouter des informations d'authentification aux AWS demandes envoyées par HTTP. Pour des raisons de sécurité, la plupart des demandes AWS doivent être signées à l'aide d'une clé d'accès, qui consiste en un identifiant de clé d'accès et une clé d'accès secrète. Ces deux clés sont généralement appelées informations d'identification de sécurité. Pour plus d'informations sur SigV4, consultez la section Processus de signature Signature Version 4.

Utilisation AWS Distro pour en OpenTelemetry tant que collectionneur

Cette section explique comment configurer le collecteur AWS Distro for OpenTelemetry (ADOT) pour qu'il soit extrait d'une Prometheus-instrumented application et envoyer les métriques à Amazon Managed Service for Prometheus. Pour plus d'informations sur l'ADOT Collector, consultez AWS Distro for. OpenTelemetry

Les rubriques suivantes décrivent trois manières différentes de configurer ADOT en tant que collecteur pour vos métriques, selon que celles-ci proviennent d'Amazon EKS, d'Amazon ECS ou d'une instance Amazon EC2.

Rubriques

- Configurez l'ingestion de métriques à l'aide de AWS Distribution pour un cluster OpenTelemetry Amazon Elastic Kubernetes Service
- Configurez l'ingestion de métriques depuis Amazon ECS à l'aide de AWS Distro pour Open Telemetry
- Configuration de l'ingestion de métriques à partir d'une instance Amazon EC2 à l'aide de l'écriture à distance

Configurez l'ingestion de métriques à l'aide de AWS Distribution pour un cluster OpenTelemetry Amazon Elastic Kubernetes Service

Vous pouvez utiliser le collecteur AWS Distro for OpenTelemetry (ADOT) pour extraire les métriques d'une Prometheus-instrumented application et les envoyer à Amazon Managed Service for Prometheus.

> **Note**
>
> Pour plus d'informations sur le collecteur ADOT, consultez AWS Distro for. OpenTelemetry Pour plus d'informations sur Prometheus-instrumented les applications, consultezQuels sont Prometheus-compatible des indicateurs ?.

La collecte de métriques Prometheus avec ADOT implique trois OpenTelemetry composants : le récepteur Prometheus, l'exportateur d'écriture à distance Prometheus et l'extension d'authentification Sigv4.

Vous pouvez configurer Prometheus Receiver à l'aide de votre configuration Prometheus existante pour effectuer la découverte de service et la collecte des métriques. Prometheus Receiver collecte des métriques dans le format d'exposition Prometheus. Toutes les applications ou points de terminaison que vous souhaitez collecter doivent être configurés avec la bibliothèque client Prometheus. Prometheus Receiver prend en charge l'ensemble complet des configurations de collecte et de réétiquetage de Prometheus décrites dans la section Configuration de la documentation

Prometheus. Vous pouvez coller ces configurations directement dans les configurations de votre collecteur ADOT.

Prometheus Remote Write Exporter utilise le point de terminaison `remote_write` pour envoyer les métriques collectées à l'espace de travail de votre portail de gestion. Les requêtes HTTP pour exporter des données seront signées avec AWS Sigv4, le AWS protocole d'authentification sécurisée, avec l'extension d'authentification Sigv4. Pour plus d'informations, consultez Processus de signature Signature Version 4.

Le collecteur reconnaît automatiquement les points de terminaison des métriques Prometheus sur Amazon EKS et utilise la configuration trouvée dans <kubernetes_sd_config>.

La démonstration suivante est un exemple de cette configuration sur un cluster exécutant Amazon Elastic Kubernetes Service ou Kubernetes autogéré. Pour effectuer ces étapes, vous devez disposer AWS d'informations d'identification provenant de l'une des options potentielles de la chaîne AWS d'informations d'identification par défaut. Pour plus d'informations, consultez la section Configuration du AWS SDK pour Go. Cette démonstration utilise un exemple d'application qui est utilisé pour les tests d'intégration du processus. L'exemple d'application expose les métriques au niveau du point de terminaison `/metrics`, comme la bibliothèque client Prometheus.

Conditions préalables

Avant de commencer les étapes de configuration d'ingestion suivantes, vous devez configurer votre rôle IAM pour le compte de service et la politique d'approbation.

Pour configurer le rôle IAM pour le compte de service et la politique d'approbation

1. Créez le rôle IAM pour le compte de service en suivant les étapes décrites dans Configuration de rôles de service pour l'ingestion de métriques à partir de clusters Amazon EKS.

   Le collecteur ADOT utilisera ce rôle lorsqu'il collectera et exportera des métriques.
2. Modifiez ensuite la politique d'approbation. Ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
3. Dans le volet de navigation de gauche, choisissez Roles et recherchez le rôle amp-iamproxy-ingest-role que vous avez créé à l'étape 1.
4. Choisissez l'onglet Relations d'approbation, puis Modifier la relation d'approbation.
5. Dans le JSON de la politique de relation d'approbation, remplacez `aws-amp` par `adot-col`, puis choisissez Update Trust Policy. La politique d'approbation obtenue doit être similaire à ce qui suit :

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Federated": "arn:aws:iam::111122223333:oidc-provider/oidc.eks.us-
east-1.amazonaws.com/id/EXAMPLED539D4633E53DE1B71EXAMPLE"
      },
      "Action": "sts:AssumeRoleWithWebIdentity",
      "Condition": {
        "StringEquals": {
          "oidc.eks.us-east-1.amazonaws.com/
id/EXAMPLED539D4633E53DE1B71EXAMPLE:sub": "system:serviceaccount:adot-
col:amp-iamproxy-ingest-service-account",
          "oidc.eks.us-east-1.amazonaws.com/
id/EXAMPLED539D4633E53DE1B71EXAMPLE:aud": "sts.amazonaws.com"
        }
      }
    }
  ]
}
```

6. Choisissez l'onglet Autorisations et assurez-vous que la politique d'autorisations suivante est associée au rôle.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "aps:RemoteWrite",
                "aps:GetSeries",
                "aps:GetLabels",
                "aps:GetMetricMetadata"
            ],
```

```
            "Resource": "*"
        }
    ]
}
```

Activation de la collecte de métriques Prometheus

> **Note**
>
> Lorsque vous créez un espace de noms dans Amazon EKS, `alertmanager` l'exportateur de nœuds sont désactivés par défaut.

Pour activer la collecte Prometheus sur un cluster Amazon EKS ou Kubernetes

1. Créez et clonez l'exemple d'application depuis le référentiel sur aws-otel-community.

   Exécutez ensuite les commandes suivantes.

   ```
   cd ./sample-apps/prometheus-sample-app
   docker build . -t prometheus-sample-app:latest
   ```

2. Transférez cette image vers un registre tel qu'Amazon ECR ou DockerHub.
3. Déployez l'exemple d'application dans le cluster en copiant cette configuration Kubernetes et en l'appliquant. Remplacez l'image par celle que vous venez d'envoyer en remplaçant `{{PUBLIC_SAMPLE_APP_IMAGE}}` dans le fichier `prometheus-sample-app.yaml`.

   ```
   curl https://raw.githubusercontent.com/aws-observability/aws-otel-collector/
   main/examples/eks/aws-prometheus/prometheus-sample-app.yaml -o prometheus-sample-
   app.yaml
   kubectl apply -f prometheus-sample-app.yaml
   ```

4. Exécutez la commande suivante pour vérifier que l'exemple d'application a démarré. Dans la sortie de la commande, `prometheus-sample-app` apparaît dans la colonne `NAME`.

   ```
   kubectl get all -n aoc-prometheus-pipeline-demo
   ```

5. Démarrez une instance par défaut du collecteur ADOT. Pour ce faire, commencez par entrer la commande suivante pour extraire la configuration Kubernetes du collecteur ADOT.

```
curl https://raw.githubusercontent.com/aws-observability/aws-otel-collector/main/
examples/eks/aws-prometheus/prometheus-daemonset.yaml -o prometheus-daemonset.yaml
```

Modifiez ensuite le fichier modèle en remplaçant le point de terminaison remote_write de votre espace de travail Amazon Managed Service for Prometheus par `YOUR_ENDPOINT` et votre région par `YOUR_REGION`. Utilisez le point de terminaison remote_write affiché dans la console Amazon Managed Service for Prometheus lorsque vous consultez les détails de votre espace de travail.

Vous devrez également modifier votre identifiant de compte `YOUR_ACCOUNT_ID` dans la section Compte de service de la configuration de Kubernetes. AWS

Dans cet exemple, la configuration du collecteur ADOT utilise une annotation (`scrape=true`) pour indiquer les points de terminaison cible à collecter. Cela permet au collecteur ADOT de distinguer le point de terminaison de l'exemple d'application des points de terminaison kube-system dans votre cluster. Vous pouvez le supprimer des configurations de réétiquetage si vous souhaitez récupérer un autre exemple d'application.

6. Entrez la commande suivante pour déployer le collecteur ADOT.

```
kubectl apply -f prometheus-daemonset.yaml
```

7. Exécutez la commande suivante pour vérifier que le collecteur ADOT a démarré. Recherchez `adot-col` dans la colonne `NAMESPACE`.

```
kubectl get pods -n adot-col
```

8. Vérifiez que le pipeline fonctionne à l'aide de l'exportateur de journalisation. Notre exemple de modèle est déjà intégré à l'exportateur de journalisation. Entrez les commandes suivantes :

```
kubectl get pods -A
kubectl logs -n adot-col name_of_your_adot_collector_pod
```

Certaines des métriques collectées de l'exemple d'application ressembleront à celles de l'exemple suivant.

```
Resource labels:
     -> service.name: STRING(kubernetes-service-endpoints)
     -> host.name: STRING(192.168.16.238)
```

```
     -> port: STRING(8080)
     -> scheme: STRING(http)
InstrumentationLibraryMetrics #0
Metric #0
Descriptor:
     -> Name: test_gauge0
     -> Description: This is my gauge
     -> Unit:
     -> DataType: DoubleGauge
DoubleDataPoints #0
StartTime: 0
Timestamp: 1606511460471000000
Value: 0.000000
```

9. Pour vérifier si Amazon Managed Service for Prometheus a reçu les statistiques, utilisez `awscurl`. Cet outil vous permet d'envoyer des requêtes HTTP via la ligne de commande avec l'authentification AWS Sigv4. Vous devez donc disposer d' AWS informations d'identification configurées localement avec les autorisations appropriées pour effectuer des requêtes auprès d'Amazon Managed Service for Prometheus. Pour les instructions d'installation`awscurl`, consultez awscurl. https://github.com/okigan/awscurl

   Dans la commande suivante, remplacez `AMP_REGION` et `AMP_ENDPOINT` par les informations relatives à votre espace de travail Amazon Managed Service for Prometheus.

   ```
   awscurl --service="aps" --region="AMP_REGION" "https://AMP_ENDPOINT/api/v1/query?
   query=adot_test_gauge0"
   {"status":"success","data":{"resultType":"vector","result":[{"metric":
   {"__name__":"adot_test_gauge0"},"value":[1606512592.493,"16.87214000011479"]}]}}
   ```

   Si vous recevez une métrique en réponse, cela signifie que la configuration de votre pipeline est réussie et que la métrique s'est propagée avec succès depuis l'exemple d'application dans Amazon Managed Service for Prometheus.

Nettoyage

Pour nettoyer cette démo, entrez les commandes suivantes.

```
kubectl delete namespace aoc-prometheus-pipeline-demo
kubectl delete namespace adot-col
```

Configuration avancée

Prometheus Receiver prend en charge l'ensemble complet des configurations de collecte et de réétiquetage de Prometheus décrites dans la section Configuration de la documentation Prometheus. Vous pouvez coller ces configurations directement dans les configurations de votre collecteur ADOT.

La configuration de Prometheus Receiver inclut vos configurations de découverte de service, de collecte et de réétiquetage. La configuration du récepteur ressemble à ce qui suit.

```
receivers:
  prometheus:
    config:
      [Your Prometheus configuration]
```

Voici un exemple de configuration.

```
receivers:
  prometheus:
    config:
      global:
        scrape_interval: 1m
        scrape_timeout: 10s

      scrape_configs:
      - job_name: kubernetes-service-endpoints
        sample_limit: 10000
        kubernetes_sd_configs:
        - role: endpoints
        tls_config:
          ca_file: /var/run/secrets/kubernetes.io/serviceaccount/ca.crt
          insecure_skip_verify: true
        bearer_token_file: /var/run/secrets/kubernetes.io/serviceaccount/token
```

Si vous disposez d'une configuration Prometheus existante, vous devez remplacer les caractères $ par les caractères $$ pour éviter que les valeurs soient remplacées par des variables d'environnement. *Ceci est particulièrement important pour la valeur de remplacement de relabel_configurations. Par exemple, si vous commencez par la configuration relabel_configuration suivante :

```
relabel_configs:
- source_labels:
 [__meta_kubernetes_ingress_scheme,__address__,__meta_kubernetes_ingress_path]
```

```
  regex: (.+);(.+);(.+)
  replacement: ${1}://${2}${3}
  target_label: __param_target
```

Elle deviendra :

```
relabel_configs:
- source_labels:
 [__meta_kubernetes_ingress_scheme,__address__,__meta_kubernetes_ingress_path]
  regex: (.+);(.+);(.+)
  replacement: $${1}://${2}${3}
  target_label: __param_target
```

Prometheus Remote Write Exporter et SigV4 Authentication Extension

Les configurations de Prometheus Remote Write Exporter et Sigv4 Authentication Extension sont plus simples que celle de Prometheus Receiver. À ce stade du pipeline, les métriques ont déjà été ingérées et nous sommes prêts à exporter ces données dans Amazon Managed Service for Prometheus. L'exemple suivant montre la configuration minimale requise pour communiquer avec Amazon Managed Service for Prometheus.

```
extensions:
  sigv4auth:
    service: "aps"
    region: "user-region"
exporters:
  prometheusremotewrite:
    endpoint: "https://aws-managed-prometheus-endpoint/api/v1/remote_write"
    auth:
      authenticator: "sigv4auth"
```

Cette configuration envoie une requête HTTPS qui est signée par AWS SIGv4 à l'aide des AWS informations d'identification de la chaîne d'informations AWS d'identification par défaut. Pour plus d'informations, consultez Configuration de l' AWS SDK pour Go. Vous devez spécifier le service `aps`.

Quelle que soit la méthode de déploiement, le collecteur ADOT doit avoir accès à l'une des options répertoriées dans la chaîne AWS d'informations d'identification par défaut. L'extension d'authentification Sigv4 dépend de AWS SDK pour Go et l'utilise pour récupérer les informations d'identification et s'authentifier. Vous devez vous assurer que ces informations d'identification disposent d'autorisations d'écriture à distance pour Amazon Managed Service for Prometheus.

Configurez l'ingestion de métriques depuis Amazon ECS à l'aide de AWS Distro pour Open Telemetry

Cette section explique comment collecter des métriques depuis Amazon Elastic Container Service (Amazon ECS) et les intégrer dans Amazon Managed Service for Prometheus à l'aide de AWS Distro for Open Telemetry (ADOT). Elle décrit également comment visualiser vos métriques dans Amazon Managed Grafana.

Conditions préalables

> **Important**
>
> Avant de commencer, vous devez disposer d'un environnement Amazon ECS sur un cluster AWS Fargate avec des paramètres par défaut, d'un espace de travail Amazon Managed Service for Prometheus et d'un espace de travail Amazon Managed Grafana. Nous supposons que vous connaissez les charges de travail liées aux conteneurs, Amazon Managed Service for Prometheus et Amazon Managed Grafana.

Pour plus d'informations, consultez les liens suivants :

- Pour plus d'informations sur la création d'un environnement Amazon ECS dans un cluster Fargate avec des paramètres par défaut, consultez la section Création d'un cluster dans le Guide du développeur Amazon ECS.
- Pour plus d'informations sur la création d'un espace de travail Amazon Managed Service for Prometheus, consultez la section Création d'un espace de travail dans le Guide de l'utilisateur Amazon Managed Service for Prometheus.
- Pour plus d'informations sur la création d'un espace de travail Amazon Managed Grafana, consultez la section Création d'un espace de travail dans le Guide de l'utilisateur Amazon Managed Grafana.

Étape 1 : définir une image de conteneur ADOT Collector personnalisée

Utilisez le fichier de configuration suivant comme modèle pour définir votre propre image de conteneur de collecteur ADOT. Remplacez *my-remote-URL* et *my-region* par `endpoint` les `region` vôtres. Enregistrez la configuration dans un fichier appelé adot-config.yaml.

> **Note**
>
> Cette configuration utilise l'extension `sigv4auth` pour authentifier les appels à Amazon Managed Service for Prometheus. Pour plus d'informations sur la configuration`sigv4auth`, consultez Authenticator - Sigv4 activé. GitHub

```
receivers:
  prometheus:
    config:
      global:
        scrape_interval: 15s
        scrape_timeout: 10s
      scrape_configs:
      - job_name: "prometheus"
        static_configs:
        - targets: [ 0.0.0.0:9090 ]
  awsecscontainermetrics:
    collection_interval: 10s
processors:
  filter:
    metrics:
      include:
        match_type: strict
        metric_names:
          - ecs.task.memory.utilized
          - ecs.task.memory.reserved
          - ecs.task.cpu.utilized
          - ecs.task.cpu.reserved
          - ecs.task.network.rate.rx
          - ecs.task.network.rate.tx
          - ecs.task.storage.read_bytes
          - ecs.task.storage.write_bytes
exporters:
  prometheusremotewrite:
    endpoint: my-remote-URL
    auth:
      authenticator: sigv4auth
  logging:
    loglevel: info
extensions:
  health_check:
```

```
  pprof:
    endpoint: :1888
  zpages:
    endpoint: :55679
  sigv4auth:
    region: my-region
    service: aps
service:
  extensions: [pprof, zpages, health_check, sigv4auth]
  pipelines:
    metrics:
      receivers: [prometheus]
      exporters: [logging, prometheusremotewrite]
    metrics/ecs:
      receivers: [awsecscontainermetrics]
      processors: [filter]
      exporters: [logging, prometheusremotewrite]
```

Étape 2 : Transférez l'image de votre conteneur ADOT Collector vers un référentiel Amazon ECR

Utilisez un fichier Dockerfile pour créer et envoyer votre image de conteneur à un référentiel Amazon Elastic Container Registry (ECR).

1. Créez le fichier Dockerfile pour copier et ajouter votre image de conteneur à l'image Docker OTEL.

   ```
   FROM public.ecr.aws/aws-observability/aws-otel-collector:latest
   COPY adot-config.yaml /etc/ecs/otel-config.yaml
   CMD ["--config=/etc/ecs/otel-config.yaml"]
   ```

2. créer un référentiel Amazon ECR ;

   ```
   # create repo:
   COLLECTOR_REPOSITORY=$(aws ecr create-repository --repository aws-otel-collector \
                                 --query repository.repositoryUri --output text)
   ```

3. Créez voter image de conteneur.

   ```
   # build ADOT collector image:
   docker build -t $COLLECTOR_REPOSITORY:ecs .
   ```

> **Note**
>
> Cela suppose que vous créez votre conteneur dans le même environnement que celui dans lequel il sera exécuté. Dans le cas contraire, vous devrez peut-être utiliser le paramètre `--platform` lors de la création de l'image.

4. Connectez-vous au référentiel Amazon ECR. Remplacez *my-region* par votre `region` valeur.

```
# sign in to repo:
aws ecr get-login-password --region my-region | \
        docker login --username AWS --password-stdin $COLLECTOR_REPOSITORY
```

5. Envoyez votre image de conteneur.

```
# push ADOT collector image:
docker push $COLLECTOR_REPOSITORY:ecs
```

Étape 3 : Création d'une définition de tâche Amazon ECS pour scraper Amazon Managed Service for Prometheus

Créez une définition de tâche Amazon ECS pour la collecte dans Amazon Managed Service for Prometheus. Votre définition de tâche doit inclure un conteneur nommé `adot-collector` et un conteneur nommé`prometheus`. `prometheus` génère des métriques et `adot-collector` effectue la collecte dans `prometheus`.

> **Note**
>
> Amazon Managed Service for Prometheus fonctionne en tant que service et collecte des métriques à partir de conteneurs. Dans ce cas, les conteneurs exécutent Prometheus localement, en mode Agent, et envoient les métriques locales à Amazon Managed Service for Prometheus.

Exemple : définition de tâche

Voici un exemple de définition de tâche. Vous pouvez utiliser cet exemple comme modèle pour créer votre propre définition de tâche. Remplacez la valeur `image` de `adot-collector` par l'URL du

référentiel et la balise d'image (`$COLLECTOR_REPOSITORY:ecs`). Remplacez les valeurs `region` de `adot-collector` et `prometheus` par vos valeurs `region`.

```
{
  "family": "adot-prom",
  "networkMode": "awsvpc",
  "containerDefinitions": [
    {
      "name": "adot-collector",
      "image": "account_id.dkr.ecr.region.amazonaws.com/image-tag",
      "essential": true,
      "logConfiguration": {
        "logDriver": "awslogs",
        "options": {
          "awslogs-group": "/ecs/ecs-adot-collector",
          "awslogs-region": "my-region",
          "awslogs-stream-prefix": "ecs",
          "awslogs-create-group": "True"
        }
      }
    },
    {
      "name": "prometheus",
      "image": "prom/prometheus:main",
      "logConfiguration": {
        "logDriver": "awslogs",
        "options": {
          "awslogs-group": "/ecs/ecs-prom",
          "awslogs-region": "my-region",
          "awslogs-stream-prefix": "ecs",
          "awslogs-create-group": "True"
        }
      }
    }
  ],
  "requiresCompatibilities": [
    "FARGATE"
  ],
  "cpu": "1024"
}
```

Étape 4 : Autorisez vos tâches à accéder à Amazon Managed Service pour Prometheus

Pour envoyer les métriques extraites à Amazon Managed Service for Prometheus, votre tâche Amazon ECS doit disposer des autorisations appropriées pour appeler les opérations d' AWS API à votre place. Vous devez créer un rôle IAM pour vos tâches et y attacher la politique `AmazonPrometheusRemoteWriteAccess`. Pour plus d'informations sur la création de ce rôle et l'attachement de cette politique, consultez la section Création d'un rôle et d'une politique IAM pour vos tâches.

Une fois que vous avez attaché `AmazonPrometheusRemoteWriteAccess` à votre rôle IAM et que vous l'avez utilisé pour vos tâches, Amazon ECS peut envoyer vos métriques collectées à Amazon Managed Service for Prometheus.

Étape 5 : visualisez vos statistiques dans Amazon Managed Grafana

> **Important**
>
> Avant de commencer, vous devez exécuter une tâche Fargate sur votre définition de tâche Amazon ECS. Sinon, Amazon Managed Service for Prometheus ne pourra pas utiliser vos métriques.

1. Dans le volet de navigation de votre espace de travail Amazon Managed Grafana, choisissez Sources de données sous l' AWS icône.
2. Dans l'onglet Sources de données, pour Service, sélectionnez Amazon Managed Service for Prometheus et choisissez votre région par défaut.
3. Choisissez Add data source.
4. Utilisez les préfixes `ecs` et `prometheus` pour interroger et visualiser vos métriques.

Configuration de l'ingestion de métriques à partir d'une instance Amazon EC2 à l'aide de l'écriture à distance

Cette section explique comment exécuter un serveur Prometheus avec l'écriture à distance dans une instance Amazon Elastic Compute Cloud (Amazon EC2). Elle explique comment collecter des métriques à partir d'une application de démonstration écrite dans Go et les envoyer à un espace de travail Amazon Managed Service for Prometheus.

Conditions préalables

> **Important**
>
> Avant de commencer, vous devez avoir installé Prometheus version 2.26 ou une version ultérieure. Nous supposons que vous connaissez Prometheus, Amazon EC2 et Amazon Managed Service for Prometheus. Pour plus d'informations sur l'installation de Prometheus, consultez la section Mise en route sur le site Web de Prometheus.

Si vous ne connaissez pas Amazon EC2 ou Amazon Managed Service for Prometheus, nous vous recommandons de commencer par lire les sections suivantes :

- Qu'est-ce qu'Amazon Elastic Compute Cloud ?
- Qu'est-ce qu'Amazon Managed Service for Prometheus ?

Création d'un rôle IAM pour Amazon EC2

Pour diffuser des métriques, vous devez d'abord créer un rôle IAM avec la politique AWS AmazonPrometheusRemoteWriteAccess gérée. Vous pouvez ensuite lancer une instance avec le rôle et les métriques de diffusion dans votre espace de travail Amazon Managed Service for Prometheus.

1. Ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Sélectionnez Rôles dans le volet de navigation, puis Créer un rôle.
3. Pour le type d'entité de confiance, choisissez service AWS . Pour le cas d'utilisation, choisissez EC2. Choisissez Suivant : Autorisations.
4. Dans la barre de recherche, saisissez AmazonPrometheusRemoteWriteAccess. Pour Nom de la politique, sélectionnez AmazonPrometheusRemoteWriteAccess, puis choisissez Joindre une politique. Sélectionnez Next:Tags.
5. (Facultatif) Créez des balises IAM pour votre rôle IAM. Choisissez Suivant : Vérification.
6. Saisissez un nom pour votre rôle. Choisissez Create Policy (Créer une politique).

Lancement d'une instance Amazon EC2

Pour lancer une instance Amazon EC2, suivez les instructions de la section Launch an instance du Guide de l'utilisateur Amazon Elastic Compute Cloud pour les instances Linux.

Exécutez l'application de démonstration.

Après avoir créé votre rôle IAM et lancé une instance EC2 avec ce rôle, vous pouvez exécuter une application de démonstration pour le voir fonctionner.

Pour exécuter une application de démonstration et tester des métriques

1. Pour créer un fichier Go nommé `main.go`, utilisez le modèle suivant.

```
package main

import (
    "github.com/prometheus/client_golang/prometheus/promhttp"
    "net/http"
)

func main() {
    http.Handle("/metrics", promhttp.Handler())

    http.ListenAndServe(":8000", nil)
}
```

2. Exécutez les commandes suivantes pour installer les bonnes dépendances.

```
sudo yum update -y
sudo yum install -y golang
go get github.com/prometheus/client_golang/prometheus/promhttp
```

3. Exécutez l'application de démonstration.

```
go run main.go
```

   L'application de démonstration doit fonctionner sur le port 8000 et afficher toutes les métriques Prometheus exposées. Voici un exemple de ces métriques.

```
curl -s http://localhost:8000/metrics
...
process_max_fds 4096# HELP process_open_fds Number of open file descriptors.# TYPE
 process_open_fds gauge
process_open_fds 10# HELP process_resident_memory_bytes Resident memory size in
 bytes.# TYPE process_resident_memory_bytes gauge
```

```
process_resident_memory_bytes 1.0657792e+07# HELP process_start_time_seconds Start
 time of the process since unix epoch in seconds.# TYPE process_start_time_seconds
 gauge
process_start_time_seconds 1.61131955899e+09# HELP process_virtual_memory_bytes
 Virtual memory size in bytes.# TYPE process_virtual_memory_bytes gauge
process_virtual_memory_bytes 7.77281536e+08# HELP process_virtual_memory_max_bytes
 Maximum amount of virtual memory available in bytes.# TYPE
 process_virtual_memory_max_bytes gauge
process_virtual_memory_max_bytes -1# HELP
 promhttp_metric_handler_requests_in_flight Current number of scrapes being
 served.# TYPE promhttp_metric_handler_requests_in_flight gauge
promhttp_metric_handler_requests_in_flight 1# HELP
 promhttp_metric_handler_requests_total Total number of scrapes by HTTP status
 code.# TYPE promhttp_metric_handler_requests_total counter
promhttp_metric_handler_requests_total{code="200"} 1
promhttp_metric_handler_requests_total{code="500"} 0
promhttp_metric_handler_requests_total{code="503"} 0
```

Création d'un espace de travail Amazon Managed Service for Prometheus

Pour créer un espace de travail Amazon Managed Service for Prometheus, suivez les instructions de la section Create a workspace.

Exécution d'un serveur Prometheus

1. Utilisez l'exemple de fichier YAML suivant comme modèle pour créer un nouveau fichier nommé `prometheus.yaml`. Pour`url`, remplacez *`my-region`* par la valeur de votre région et *`my-workspace-id`* par l'identifiant de l'espace de travail qu'Amazon Managed Service for Prometheus a généré pour vous. Pour`region`, remplacez *`my-region`* par la valeur de votre région.

   Exemple : fichier YAML

   ```
   global:
     scrape_interval: 15s
     external_labels:
       monitor: 'prometheus'

   scrape_configs:
     - job_name: 'prometheus'
       static_configs:
         - targets: ['localhost:8000']
   ```

```
remote_write:
  -
    url: https://aps-workspaces.my-region.amazonaws.com/workspaces/my-workspace-id/
api/v1/remote_write
    queue_config:
        max_samples_per_send: 1000
        max_shards: 200
        capacity: 2500
    sigv4:
         region: my-region
```

2. Exécutez le serveur Prometheus pour envoyer les métriques de l'application de démonstration à votre espace de travail Amazon Managed Service for Prometheus.

```
prometheus --config.file=prometheus.yaml
```

Le serveur Prometheus doit maintenant envoyer les métriques de l'application de démonstration à votre espace de travail Amazon Managed Service for Prometheus.

Utilisation d'une instance Prometheus comme collecteur

Vous pouvez utiliser une instance de Prometheus, exécutée en mode agent (appelée agent Prometheus), pour extraire des métriques et les envoyer à votre espace de travail Amazon Managed Service for Prometheus.

Les rubriques suivantes décrivent les différentes manières de configurer une instance Prometheus exécutée en mode agent en tant que collecteur pour vos métriques.

> **Warning**
>
> Lorsque vous créez un agent Prometheus, vous êtes responsable de sa configuration et de sa maintenance. Évitez d'exposer les terminaux Prometheus Scrape à l'Internet public en activant les fonctionnalités de sécurité.

Si vous avez configuré plusieurs instances Prometheus qui surveillent le même ensemble de métriques et que vous les avez envoyées à un seul espace de travail Amazon Managed Service for Prometheus à des fins de haute disponibilité, vous devez configurer la déduplication. Si vous

ne suivez pas les étapes de configuration de la déduplication, tous les échantillons de données envoyés à Amazon Managed Service for Prometheus vous seront facturés, y compris les échantillons en double. Pour obtenir des instructions sur la configuration de la déduplication, consultez la section Déduplication des métriques haute disponibilité envoyées à Amazon Managed Service for Prometheus.

Rubriques

- Configuration de l'ingestion à partir d'un nouveau serveur Prometheus à l'aide de Helm
- Configuration de l'ingestion depuis un serveur Prometheus existant dans Kubernetes sur EC2
- Configuration de l'ingestion depuis un serveur Prometheus existant dans Kubernetes sur Fargate

Configuration de l'ingestion à partir d'un nouveau serveur Prometheus à l'aide de Helm

Les instructions de cette section vous permettent d'être rapidement opérationnel avec Amazon Managed Service for Prometheus. Vous configurez un nouveau serveur Prometheus dans un cluster Amazon EKS, et ce nouveau serveur utilise une configuration par défaut pour envoyer des métriques à Amazon Managed Service for Prometheus. Voici les prérequis pour cette méthode :

- Vous devez disposer d'un cluster Amazon EKS à partir duquel le nouveau serveur Prometheus collectera des métriques.
- Un pilote Amazon EBS CSI doit être installé sur votre cluster Amazon EKS (requis par Helm).
- Vous devez utiliser Helm CLI 3.0 ou version ultérieure.
- Vous devez utiliser un ordinateur Linux ou macOS pour effectuer les étapes décrites dans les sections suivantes.

Étape 1 : Ajouter de nouveaux référentiels de Charts Helm

Pour ajouter de nouveaux référentiels de Charts de Helm, entrez les commandes suivantes. Pour plus d'informations sur l'utilisation de ces commandes, consultez la section Helm Repo.

```
helm repo add prometheus-community https://prometheus-community.github.io/helm-charts
helm repo add kube-state-metrics https://kubernetes.github.io/kube-state-metrics
helm repo update
```

Étape 2 : Créer un espace de noms Prometheus

Entrez la commande suivante pour créer un espace de noms Prometheus pour le serveur Prometheus et les autres composants de surveillance. *prometheus-namespace*Remplacez-le par le nom que vous souhaitez attribuer à cet espace de noms.

```
kubectl create namespace prometheus-namespace
```

Étape 3 : Configurer des rôles IAM pour les comptes de service

Pour cette méthode d'intégration indiquée, vous devez utiliser des rôles IAM pour les comptes de service du cluster Amazon EKS où le serveur Prometheus est exécuté.

Avec les rôles IAM pour les comptes de service, vous pouvez associer un rôle IAM à un compte de service Kubernetes. Ce compte de service peut ensuite fournir des autorisations AWS aux pods de n'importe quel pod qui utilise ce compte de service. Pour plus d'informations, consultez la section Rôles IAM pour les comptes de service.

Si vous n'avez pas encore configuré ces rôles, suivez les instructions de la section Configuration de rôles de service pour l'ingestion de métriques à partir de clusters Amazon EKS pour les configurer. Les instructions de cette section nécessitent l'utilisation de `eksctl`. Pour plus d'informations, consultez la section Démarrer avec Amazon Elastic Kubernetes Service – `eksctl`.

> **Note**
>
> Lorsque vous n'êtes pas sur EKS ou AWS que vous utilisez simplement une clé d'accès et une clé secrète pour accéder à Amazon Managed Service for Prometheus, vous ne pouvez pas utiliser le SIGv4 `EKS-IAM-ROLE` basé sur la base.

Étape 4 : Configurer le nouveau serveur et commencer à ingérer des métriques

Pour installer le nouveau serveur Prometheus qui envoie des métriques à votre espace de travail Amazon Managed Service for Prometheus, procédez comme suit.

Pour installer un nouveau serveur Prometheus afin d'envoyer des métriques à votre espace de travail Amazon Managed Service for Prometheus

1. À l'aide d'un éditeur de texte, créez un fichier nommé `my_prometheus_values_yaml` avec le contenu suivant.

- Remplacez *IAM_PROXY_PROMETHEUS_ROLE_ARN* par l'ARN du rôle amp-iamproxy-ingest-role créé dans Configuration de rôles de service pour l'ingestion de métriques à partir de clusters Amazon EKS.
- Remplacez-le *WORKSPACE_ID* par l'ID de votre espace de travail Amazon Managed Service pour Prometheus.
- *REGION*Remplacez-la par la région de votre espace de travail Amazon Managed Service for Prometheus.

```
## The following is a set of default values for prometheus server helm chart which
 enable remoteWrite to AMP
## For the rest of prometheus helm chart values see: https://github.com/prometheus-
community/helm-charts/blob/main/charts/prometheus/values.yaml
##
serviceAccounts:
  server:
    name: amp-iamproxy-ingest-service-account
    annotations:
      eks.amazonaws.com/role-arn: ${IAM_PROXY_PROMETHEUS_ROLE_ARN}
server:
  remoteWrite:
    - url: https://aps-workspaces.${REGION}.amazonaws.com/workspaces/
${WORKSPACE_ID}/api/v1/remote_write
      sigv4:
        region: ${REGION}
      queue_config:
        max_samples_per_send: 1000
        max_shards: 200
        capacity: 2500
```

2. Saisissez la commande suivante pour créer le serveur Prometheus.

- Remplacez *prometheus-chart-name* par le nom de votre version de Prometheus.
- Remplacez-le *prometheus-namespace* par le nom de votre espace de noms Prometheus.

```
helm install prometheus-chart-name prometheus-community/prometheus -n prometheus-
namespace \
-f my_prometheus_values_yaml
```

> **Note**
>
> Vous pouvez personnaliser la commande `helm install` de différentes façons. Pour plus d'informations, consultez Installation de Helm dans la documentation Helm.

Configuration de l'ingestion depuis un serveur Prometheus existant dans Kubernetes sur EC2

Amazon Managed Service for Prometheus prend en charge l'ingestion de métriques à partir de serveurs Prometheus dans des clusters exécutés sur Amazon EKS et dans des clusters Kubernetes exécutés sur Amazon EC2. Les instructions détaillées de cette section concernent un serveur Prometheus dans un cluster Amazon EKS. Les étapes pour un cluster Kubernetes autogéré sur Amazon EC2 sont les mêmes, sauf que vous devrez configurer vous-même le fournisseur OIDC et les rôles IAM pour les comptes de service dans le cluster Kubernetes.

Les instructions de cette section utilisent Helm comme gestionnaire de packages Kubernetes.

Rubriques

- Étape 1 : Configurer des rôles IAM pour les comptes de service
- Étape 2 : Mettre à niveau votre serveur Prometheus existant à l'aide de Helm

Étape 1 : Configurer des rôles IAM pour les comptes de service

Pour cette méthode d'intégration indiquée, vous devez utiliser des rôles IAM pour les comptes de service du cluster Amazon EKS où le serveur Prometheus est exécuté. Ces rôles sont également appelés fonctions du service.

Avec les rôles de service, vous pouvez associer un rôle IAM à un compte de service Kubernetes. Ce compte de service peut ensuite fournir des AWS autorisations aux conteneurs de n'importe quel pod qui utilise ce compte de service. Pour plus d'informations, consultez la section Rôles IAM pour les comptes de service.

Si vous n'avez pas encore configuré ces rôles, suivez les instructions de la section Configuration de rôles de service pour l'ingestion de métriques à partir de clusters Amazon EKS pour les configurer.

Étape 2 : Mettre à niveau votre serveur Prometheus existant à l'aide de Helm

Les instructions de cette section incluent la configuration de l'écriture à distance et de sigv4 pour authentifier et autoriser le serveur Prometheus à écrire à distance sur votre espace de travail Amazon Managed Service for Prometheus.

Utilisation de Prometheus version 2.26.0 ou ultérieure

Suivez ces étapes si vous utilisez des Charts de Helm avec une image du serveur Prometheus version 2.26.0 ou ultérieure.

Pour configurer l'écriture à distance depuis un serveur Prometheus à l'aide de Charts de Helm

1. Créez une nouvelle section d'écriture à distance dans votre fichier de configuration Helm :

   - Remplacez `${IAM_PROXY_PROMETHEUS_ROLE_ARN}` par l'ARN du rôle amp-iamproxy-ingest-role créé dans Étape 1 : Configurer des rôles IAM pour les comptes de service. L'ARN du rôle doit être au format `arn:aws:iam::`*`your account ID`*`:role/amp-iamproxy-ingest-role`.
   - Remplacez `${WORKSPACE_ID}` par l'ID de votre espace de travail Amazon Managed Service for Prometheus.
   - Remplacez `${REGION}` par la Région de votre espace de travail Amazon Managed Service for Prometheus (comme `us-west-2`).

```
## The following is a set of default values for prometheus server helm chart which
 enable remoteWrite to AMP
    ## For the rest of prometheus helm chart values see: https://github.com/
prometheus-community/helm-charts/blob/main/charts/prometheus/values.yaml
    ##
    serviceAccounts:
      server:
        name: amp-iamproxy-ingest-service-account
        annotations:
          eks.amazonaws.com/role-arn: ${IAM_PROXY_PROMETHEUS_ROLE_ARN}
    server:
      remoteWrite:
        - url: https://aps-workspaces.${REGION}.amazonaws.com/workspaces/
${WORKSPACE_ID}/api/v1/remote_write
          sigv4:
            region: ${REGION}
```

```
          queue_config:
            max_samples_per_send: 1000
            max_shards: 200
            capacity: 2500
```

2. Mettez à jour la configuration existante de votre serveur Prometheus à l'aide de Helm :

   - Remplacez `prometheus-chart-name` par le nom de votre version de Prometheus.
   - Remplacez `prometheus-namespace` par l'espace de noms Kubernetes dans lequel votre serveur Prometheus est installé.
   - Remplacez `my_prometheus_values_yaml` par le chemin d'accès à votre fichier de configuration Helm.
   - Remplacez `current_helm_chart_version` par la version actuelle de vos Charts de Helm du serveur Prometheus. Vous pouvez trouver la version actuelle du graphique à l'aide de la commande helm list.

```
helm upgrade prometheus-chart-name prometheus-community/prometheus \
       -n prometheus-namespace \
       -f my_prometheus_values_yaml \
       --version current_helm_chart_version
```

Utilisation de versions antérieures de Prometheus

Suivez ces étapes si vous utilisez une version de Prometheus antérieure à la version 2.26.0. Ces étapes utilisent une approche parallèle, car les versions précédentes de Prometheus ne prenaient pas en charge nativement le processus de AWS signature Signature Version 4 (Sigv4).AWS

Ces instructions supposent que vous utilisez Helm pour déployer Prometheus.

Pour configurer l'écriture à distance depuis un serveur Prometheus

1. Sur votre serveur Prometheus, créez une nouvelle configuration d'écriture à distance. Commencez par créer un nouveau fichier de mise à jour. Nous appellerons ce fichier `amp_ingest_override_values.yaml`.

   Ajoutez les valeurs suivantes au fichier YALM.

```
serviceAccounts:
        server:
```

```
            name: "amp-iamproxy-ingest-service-account"
            annotations:
                eks.amazonaws.com/role-arn:
 "${SERVICE_ACCOUNT_IAM_INGEST_ROLE_ARN}"
    server:
        sidecarContainers:
            - name: aws-sigv4-proxy-sidecar
              image: public.ecr.aws/aws-observability/aws-sigv4-proxy:1.0
              args:
              - --name
              - aps
              - --region
              - ${REGION}
              - --host
              - aps-workspaces.${REGION}.amazonaws.com
              - --port
              - :8005
              ports:
              - name: aws-sigv4-proxy
                containerPort: 8005
        statefulSet:
            enabled: "true"
        remoteWrite:
            - url: http://localhost:8005/workspaces/${WORKSPACE_ID}/api/v1/
remote_write
```

Remplacez `${REGION}` par la Région de votre espace de travail Amazon Managed Service for Prometheus.

Remplacez `${SERVICE_ACCOUNT_IAM_INGEST_ROLE_ARN}` par l'ARN du rôle amp-iamproxy-ingest-role créé dans Étape 1 : Configurer des rôles IAM pour les comptes de service. L'ARN du rôle doit être au format `arn:aws:iam::`*`your account ID`*`:role/amp-iamproxy-ingest-role`.

Remplacez `${WORKSPACE_ID}` par votre ID d'espace de travail.

2. Mettez à niveau vos Charts de Helm Prometheus. Commencez par rechercher le nom de vos Charts de Helm en entrant la commande suivante. Dans la sortie de cette commande, recherchez un graphique dont le nom inclut `prometheus`.

```
helm ls --all-namespaces
```

Entrez ensuite la commande suivante.

```
helm upgrade --install prometheus-helm-chart-name prometheus-community/prometheus -
n prometheus-namespace -f ./amp_ingest_override_values.yaml
```

*prometheus-helm-chart-name*Remplacez-le par le nom de l'organigramme de Prometheus renvoyé dans la commande précédente. Remplacez *prometheus-namespace* par le nom de votre espace de noms.

Téléchargement de Charts de Helm

Si vous n'avez pas encore téléchargé les Charts de Helm en local, vous pouvez utiliser la commande suivante pour les télécharger.

```
helm repo add prometheus-community https://prometheus-community.github.io/helm-charts
    helm pull prometheus-community/prometheus --untar
```

Configuration de l'ingestion depuis un serveur Prometheus existant dans Kubernetes sur Fargate

Amazon Managed Service for Prometheus prend en charge l'ingestion de métriques à partir de serveurs Prometheus dans des clusters Kubernetes autogérés exécutés sur Fargate. Pour ingérer des métriques depuis des serveurs Prometheus dans des clusters Amazon EKS exécutés sur Fargate, remplacez les configurations par défaut dans un fichier de configuration nommé amp_ingest_override_values.yaml comme suit :

```
prometheus-node-exporter:
  enabled: false

alertmanager:
  enabled: false

serviceAccounts:
  server:
    name: amp-iamproxy-ingest-service-account
    annotations:
      eks.amazonaws.com/role-arn: ${IAM_PROXY_PROMETHEUS_ROLE_ARN}

server:
```

```
  persistentVolume:
    enabled: false
  remoteWrite:
    - url: https://aps-workspaces.${REGION}.amazonaws.com/workspaces/${WORKSPACE_ID}/
api/v1/remote_write
      sigv4:
        region: ${REGION}
      queue_config:
        max_samples_per_send: 1000
        max_shards: 200
        capacity: 2500
```

Installez Prometheus en utilisant les remplacements avec la commande suivante :

```
helm install prometheus-for-amp prometheus-community/prometheus \
                  -n prometheus \
                  -f amp_ingest_override_values.yaml
```

Notez que dans la configuration des Charts de Helm, nous avons désactivé l'exportateur de nœuds et le gestionnaire d'alertes, ainsi que le déploiement du serveur Prometheus.

Vous pouvez vérifier l'installation à l'aide de l'exemple de requête de test suivant.

```
$ awscurl --region region --service aps "https://aps-
workspaces.region_id.amazonaws.com/workspaces/workspace_id/api/v1/query?
query=prometheus_api_remote_read_queries"
Output:
    {"status":"success","data":{"resultType":"vector","result":[{"metric":
{"__name__":"prometheus_api_remote_read_queries","instance":"localhost:9090","job":"prometheus"
[1648461236.419,"0"]}]}}21
```

Configurez Amazon Managed Service pour Prometheus afin d'obtenir des données à haute disponibilité

Lorsque vous envoyez des données à Amazon Managed Service for Prometheus, elles sont automatiquement répliquées dans les zones de disponibilité AWS de la région et vous sont proposées à partir d'un cluster d'hôtes, garantissant l'évolutivité, la disponibilité et la sécurité. Vous souhaiterez peut-être ajouter des dispositifs de sécurité haute disponibilité supplémentaires, en fonction de votre configuration particulière. Il existe deux méthodes courantes pour ajouter des mesures de sécurité haute disponibilité à votre configuration :

- Si plusieurs conteneurs ou instances contiennent les mêmes données, vous pouvez envoyer ces données à Amazon Managed Service for Prometheus et les dédupliquer automatiquement. Cela permet de garantir que vos données seront envoyées à votre espace de travail Amazon Managed Service for Prometheus.

  Pour plus d'informations sur la déduplication des données haute disponibilité, consultez la section Déduplication des métriques haute disponibilité envoyées à Amazon Managed Service for Prometheus.
- Si vous souhaitez vous assurer l'accès à vos données, même lorsque la région AWS n'est pas disponible, vous pouvez envoyer vos métriques à un deuxième espace de travail, dans une autre région.

  Pour plus d'informations sur l'envoi de données de métriques à plusieurs espaces de travail, consultez la section Utilisez des espaces de travail interrégionaux pour améliorer la haute disponibilité dans Amazon Managed Service for Prometheus.

Rubriques

- Déduplication des métriques haute disponibilité envoyées à Amazon Managed Service for Prometheus
- Envoi de données haute disponibilité à Amazon Managed Service for Prometheus avec Prometheus
- Configurez les données de haute disponibilité dans Amazon Managed Service pour Prometheus à l'aide du graphique Prometheus Operator Helm
- Envoyez des données de haute disponibilité à Amazon Managed Service for Prometheus avec AWS Distro pour OpenTelemetry
- Envoi de données haute disponibilité à Amazon Managed Service for Prometheus avec les Charts de Helm de la communauté Prometheus
- Réponses aux questions courantes concernant la configuration de haute disponibilité dans Amazon Managed Service pour Prometheus
- Utilisez des espaces de travail interrégionaux pour améliorer la haute disponibilité dans Amazon Managed Service for Prometheus

Déduplication des métriques haute disponibilité envoyées à Amazon Managed Service for Prometheus

Vous pouvez envoyer des données provenant de plusieurs agents Prometheus (instances Prometheus exécutées en mode Agent) à votre espace de travail Amazon Managed Service for Prometheus. Si certaines de ces instances enregistrent et envoient les mêmes métriques, la disponibilité de vos données sera plus élevée (même si l'un des agents arrête d'envoyer des données, l'espace de travail Amazon Managed Service for Prometheus continuera de recevoir les données d'une autre instance). Cependant, vous souhaitez que votre espace de travail Amazon Managed Service for Prometheus déduplique automatiquement les métriques afin de ne pas les voir plusieurs fois et de ne pas être facturé pour l'ingestion et le stockage des données à plusieurs reprises.

Pour qu'Amazon Managed Service for Prometheus déduplique automatiquement les données de plusieurs agents Prometheus, vous devez attribuer à l'ensemble des agents qui envoient les données en double un nom de cluster unique, et à chacune des instances un nom de réplique. Le nom du cluster identifie les instances comme ayant des données partagées, et le nom de la réplique permet à Amazon Managed Service for Prometheus d'identifier la source de chaque métrique. Les métriques finales stockées incluent l'étiquette du cluster, mais pas la réplique. Elles semblent donc provenir d'une source unique.

> **Note**
>
> Certaines versions de Kubernetes (1.28 et 1.29) peuvent émettre leur propre métrique avec une étiquette. `cluster` Cela peut entraîner des problèmes avec la déduplication d'Amazon Managed Service for Prometheus. Consultez la FAQ sur la haute disponibilité pour plus d'informations.

Les rubriques suivantes expliquent comment envoyer des données et incluent les `__replica__` libellés `cluster` et, afin qu'Amazon Managed Service for Prometheus déduplique automatiquement les données.

> **Important**
>
> Si vous ne configurez pas la déduplication, tous les échantillons de données envoyés à Amazon Managed Service for Prometheus vous seront facturés. Ces échantillons de données incluent les échantillons en double.

Envoi de données haute disponibilité à Amazon Managed Service for Prometheus avec Prometheus

Pour définir une configuration haute disponibilité avec Prometheus, vous devez appliquer des étiquettes externes sur toutes les instances d'un groupe haute disponibilité, afin qu'Amazon Managed Service for Prometheus puisse les identifier. Utilisez l'étiquette `cluster` pour identifier un agent d'instance Prometheus dans un groupe haute disponibilité. Utilisez l'étiquette `__replica__` pour identifier chaque réplique du groupe séparément. Vous devez appliquer les étiquettes `__replica__` et `cluster` pour que la déduplication fonctionne.

> **Note**
>
> L'étiquette `__replica__` est formatée avec deux symboles de soulignement avant et après le mot `replica`.

Exemple : extraits de code

Dans les extraits de code suivants, l'étiquette `cluster` identifie l'agent d'instance Prometheus `prom-team1` et l'étiquette `_replica_` identifie les répliques `replica1` et `replica2`.

```
cluster: prom-team1
__replica__: replica1
```

```
cluster: prom-team1
__replica__: replica2
```

Comme Amazon Managed Service for Prometheus stocke des échantillons de données provenant de répliques haute disponibilité avec ces étiquettes, il retire l'étiquette `replica` lorsque les échantillons sont acceptés. Cela signifie que vous n'aurez qu'un mappage de série 1:1 pour votre série actuelle au lieu d'une série par réplique. L'étiquette `cluster` est conservée.

> **Note**
>
> Certaines versions de Kubernetes (1.28 et 1.29) peuvent émettre leur propre métrique avec une étiquette. `cluster` Cela peut entraîner des problèmes avec la déduplication d'Amazon Managed Service for Prometheus. Consultez la FAQ sur la haute disponibilité pour plus d'informations.

Configurez les données de haute disponibilité dans Amazon Managed Service pour Prometheus à l'aide du graphique Prometheus Operator Helm

Pour configurer une configuration de haute disponibilité avec l'opérateur Prometheus dans Helm, vous devez appliquer des étiquettes externes à toutes les instances d'un groupe de haute disponibilité, afin qu'Amazon Managed Service for Prometheus puisse les identifier. Vous devez également définir les attributs `replicaExternalLabelName` et `externalLabels` sur les Charts de Helm de l'opérateur Prometheus.

Exemple : en-tête YAML

Dans l'en-tête YAML suivant, `cluster` est ajouté à `externalLabel` pour identifier un agent d'instance Prometheus dans le cadre d'un groupe haute disponibilité, et `replicaExternalLabels` identifie chaque réplique du groupe.

```
replicaExternalLabelName: __replica__
externalLabels:
cluster: prom-dev
```

> **Note**
>
> Certaines versions de Kubernetes (1.28 et 1.29) peuvent émettre leur propre métrique avec une étiquette. `cluster` Cela peut entraîner des problèmes avec la déduplication d'Amazon Managed Service for Prometheus. Consultez la FAQ sur la haute disponibilité pour plus d'informations.

Envoyez des données de haute disponibilité à Amazon Managed Service for Prometheus avec AWS Distro pour OpenTelemetry

AWS Distro for OpenTelemetry (ADOT) est une distribution sécurisée et prête à la production du projet. OpenTelemetry ADOT fournit des API, des bibliothèques et des agents source, afin que vous puissiez collecter des traces et des métriques distribuées pour la surveillance des applications. Pour plus d'informations sur ADOT, consultez la section À propos de AWS Distro for Open Telemetry.

Pour configurer ADOT avec une configuration à haute disponibilité, vous devez configurer une image de conteneur ADOT Collector et appliquer les étiquettes externes et à l'exportateur d'`cluster`écriture `__replica__` à distance AWS Prometheus. Cet exportateur envoie vos métriques collectées à votre espace de travail Amazon Managed Service for Prometheus via le point de terminaison `remote_write`. Lorsque vous définissez ces étiquettes sur l'exportateur d'écriture à distance, vous empêchez la conservation des métriques en double lors de l'exécution des répliques redondantes. Pour plus d'informations sur l'exportateur d'écriture à distance AWS Prometheus, voir Démarrage avec l'exportateur d'écriture à distance Prometheus pour Amazon Managed Service for Prometheus.

> **Note**
>
> Certaines versions de Kubernetes (1.28 et 1.29) peuvent émettre leur propre métrique avec une étiquette. `cluster` Cela peut entraîner des problèmes avec la déduplication d'Amazon Managed Service for Prometheus. Consultez la FAQ sur la haute disponibilité pour plus d'informations.

Envoi de données haute disponibilité à Amazon Managed Service for Prometheus avec les Charts de Helm de la communauté Prometheus

Pour définir une configuration haute disponibilité avec les Charts de Helm de la communauté Prometheus, vous devez appliquer des étiquettes externes sur toutes les instances d'un groupe haute disponibilité, afin qu'Amazon Managed Service for Prometheus puisse les identifier. Voici un exemple de la manière dont vous pouvez ajouter les `external_labels` à une seule instance de Prometheus à partir des Charts de Helm de la communauté Prometheus.

```
server:
global:
  external_labels:
      cluster: monitoring-cluster
```

```
    __replica__: replica-1
```

> **Note**
>
> Si vous souhaitez avoir plusieurs répliques, vous devez déployer le graphique plusieurs fois avec différentes valeurs de répliques, car les Charts de Helm de la communauté Prometheus ne permettent pas de définir dynamiquement la valeur de la réplique lorsque vous augmentez le nombre de répliques directement à partir du groupe de contrôleurs. Si vous préférez que l'étiquette `replica` soit définie automatiquement, utilisez les Charts de Helm prometheus-operator.

> **Note**
>
> Certaines versions de Kubernetes (1.28 et 1.29) peuvent émettre leur propre métrique avec une étiquette. `cluster` Cela peut entraîner des problèmes avec la déduplication d'Amazon Managed Service for Prometheus. Consultez la FAQ sur la haute disponibilité pour plus d'informations.

Réponses aux questions courantes concernant la configuration de haute disponibilité dans Amazon Managed Service pour Prometheus

Dois-je inclure la valeur __réplica__ dans une autre étiquette pour suivre les points d'échantillonnage ?

Dans un environnement haute disponibilité, Amazon Managed Service for Prometheus garantit que les échantillons de données ne sont pas dupliqués en élisant un leader dans le cluster d'instances Prometheus. Si la réplique leader cesse d'envoyer des échantillons de données pendant 30 secondes, Amazon Managed Service for Prometheus transforme automatiquement une autre instance de Prometheus en réplique leader et ingère les données du nouveau leader, y compris les données manquantes. Par conséquent, la réponse est non, ce n'est pas recommandé. Cela peut entraîner des problèmes tels que les suivants :

- L'interrogation d'un `count` dans ProMQL peut renvoyer une valeur supérieure à celle attendue pendant la période d'élection d'un nouveau leader.
- Le nombre de `active series` augmente pendant la période d'élection d'un nouveau leader et il atteint les `active series limits`. Pour plus d'informations, consultez la section AMP Quotas.

Kubernetes semble avoir son propre cluster label, et ne déduplique pas mes statistiques. Comment corriger ce problème ?

Une nouvelle métrique `apiserver_storage_size_bytes` a été introduite dans Kubernetes 1.28, avec une étiquette. `cluster` Cela peut entraîner des problèmes de déduplication dans Amazon Managed Service for Prometheus, qui dépendent de l'étiquette. `cluster` Dans Kubernetes 1.3, l'étiquette est renommée en `storage-cluster_id` (elle est également renommée dans les correctifs ultérieurs de 1.28 et 1.29). Si votre cluster émet cette métrique avec l'`cluster`étiquette, Amazon Managed Service for Prometheus ne peut pas déduire la série chronologique associée. Nous vous recommandons de mettre à niveau votre cluster Kubernetes vers la dernière version corrigée pour éviter ce problème. Vous pouvez également renommer l'`cluster`étiquette de votre `apiserver_storage_size_bytes` métrique avant de l'intégrer dans Amazon Managed Service for Prometheus.

> **Note**
>
> Pour plus de détails sur la modification apportée à Kubernetes, consultez Renommer le cluster Label en storage_cluster_id pour la métrique apiserver_storage_size_bytes dans le projet Kubernetes. GitHub

Utilisez des espaces de travail interrégionaux pour améliorer la haute disponibilité dans Amazon Managed Service for Prometheus

Pour ajouter la disponibilité interrégionale à vos données, vous pouvez envoyer des mesures à plusieurs espaces de travail répartis dans AWS différentes régions. Prometheus prend en charge à la fois les dispositifs d'écriture et l'écriture entre régions.

L'exemple suivant montre comment configurer un serveur Prometheus exécuté en mode Agent pour envoyer des métriques à deux espaces de travail dans différentes régions grâce à Helm.

```
extensions:
      sigv4auth:
        service: "aps"

    receivers:
      prometheus:
        config:
          scrape_configs:
```

```
            - job_name: 'kubernetes-kubelet'
              scheme: https
              tls_config:
                ca_file: /var/run/secrets/kubernetes.io/serviceaccount/ca.crt
                insecure_skip_verify: true
              bearer_token_file: /var/run/secrets/kubernetes.io/serviceaccount/token
              kubernetes_sd_configs:
              - role: node
              relabel_configs:
              - action: labelmap
                regex: __meta_kubernetes_node_label_(.+)
              - target_label: __address__
                replacement: kubernetes.default.svc.cluster.local:443
              - source_labels: [__meta_kubernetes_node_name]
                regex: (.+)
                target_label: __metrics_path__
                replacement: /api/v1/nodes/$${1}/proxy/metrics

    exporters:
      prometheusremotewrite/one:
        endpoint: "https://aps-workspaces.workspace_1_region.amazonaws.com/workspaces/
ws-workspace_1_id/api/v1/remote_write"
        auth:
          authenticator: sigv4auth
      prometheusremotewrite/two:
        endpoint: "https://aps-workspaces.workspace_2_region.amazonaws.com/workspaces/
ws-workspace_2_id/api/v1/remote_write"
        auth:
          authenticator: sigv4auth

    service:
      extensions: [sigv4auth]
      pipelines:
        metrics/one:
          receivers: [prometheus]
          exporters: [prometheusremotewrite/one]
        metrics/two:
          receivers: [prometheus]
          exporters: [prometheusremotewrite/two]
```

Interroger vos métriques Prometheus

Maintenant que les métriques sont ingérées dans l'espace de travail, vous pouvez les interroger.

Pour créer des tableaux de bord avec des représentations visuelles de vos indicateurs, vous pouvez utiliser un service tel qu'Amazon Managed Grafana. Amazon Managed Grafana (ou une instance autonome de Grafana) peut créer une interface graphique qui affiche vos statistiques dans une grande variété de styles de présentation. Pour plus d'informations sur Amazon Managed Grafana, consultez le guide de l'utilisateur d'Amazon Managed Grafana.

Vous pouvez également créer des requêtes ponctuelles, explorer vos données ou créer vos propres applications qui utilisent vos métriques à l'aide de requêtes directes. Les requêtes directes utilisent l'API Amazon Managed Service for Prometheus et le langage de requête standard Prometheus, ProMQL, pour obtenir des données depuis votre espace de travail Prometheus. Pour plus d'informations sur ProMQL et sa syntaxe, consultez la section Querying Prometheus dans la documentation Prometheus.

Rubriques

- Aide-mémoire ProMQL
- Sélecteurs de base
- Sélecteurs vectoriels de plage
- Opérateurs d'agrégation
- Fonctions courantes
- Opérateurs binaires
- Exemples de requêtes pratiques
- Sécurisez vos requêtes métriques
- Configuration d'Amazon Managed Grafana pour une utilisation avec Amazon Managed Service for Prometheus
- Configuration de Grafana open source ou Grafana Enterprise pour une utilisation avec Amazon Managed Service for Prometheus
- Requête à l'aide de Grafana exécutée dans un cluster Amazon EKS
- Requête à l'aide d' Prometheus-compatible API
- Obtenez des statistiques sur l'utilisation de vos requêtes pour chaque requête

Aide-mémoire ProMQL

Utilisez cette aide-mémoire ProMQL (Prometheus Query Language) comme référence rapide lorsque vous recherchez des métriques dans votre espace de travail Amazon Managed Service for Prometheus. Avec ProMQL, vous pouvez sélectionner et agréger des données de séries chronologiques en temps réel grâce à son langage de requête fonctionnel.

Pour plus de détails sur ProMQL, consultez ProMQL Cheat Sheet sur le site Web. PromLabs

Sélecteurs de base

Sélectionnez les séries chronologiques par nom de métrique et par étiquette correspondante :

```
# Select all time series with the metric name http_requests_total
http_requests_total

# Select time series with specific label values
http_requests_total{job="prometheus", method="GET"}

# Use label matchers
http_requests_total{status_code!="200"}          # Not equal
http_requests_total{status_code=~"2.."}          # Regex match
http_requests_total{status_code!~"4.."}          # Negative regex match
```

Sélecteurs vectoriels de plage

Sélectionnez une gamme d'échantillons au fil du temps :

```
# Select 5 minutes of data
http_requests_total[5m]

# Time units: s (seconds), m (minutes), h (hours), d (days), w (weeks), y (years)
cpu_usage[1h]
memory_usage[30s]
```

Opérateurs d'agrégation

Données agrégées sur plusieurs séries chronologiques :

```
# Sum all values
sum(http_requests_total)

# Sum by specific labels
sum by (job) (http_requests_total)
sum without (instance) (http_requests_total)

# Other aggregation operators
avg(cpu_usage)                    # Average
min(response_time)                # Minimum
max(response_time)                # Maximum
count(up)                         # Count of series
stddev(cpu_usage)                 # Standard deviation
```

Fonctions courantes

Appliquez des fonctions pour transformer vos données :

```
# Rate of increase per second (for counters)
rate(http_requests_total[5m])

# Increase over time range
increase(http_requests_total[1h])

# Derivative (for gauges)
deriv(cpu_temperature[5m])

# Mathematical functions
abs(cpu_usage - 50)               # Absolute value
round(cpu_usage, 0.1)             # Round to nearest 0.1
sqrt(memory_usage)                # Square root

# Time functions
time()                            # Current Unix timestamp
hour()                            # Hour of day (0-23)
```

```
day_of_week()                    # Day of week (0-6, Sunday=0)
```

Opérateurs binaires

Effectuez des opérations arithmétiques et logiques :

```
# Arithmetic operators
cpu_usage + 10
memory_total - memory_available
disk_usage / disk_total * 100

# Comparison operators (return 0 or 1)
cpu_usage > 80
memory_usage < 1000
response_time >= 0.5

# Logical operators
(cpu_usage > 80) and (memory_usage > 1000)
(status_code == 200) or (status_code == 201)
```

Exemples de requêtes pratiques

Requêtes de surveillance courantes que vous pouvez utiliser dans votre espace de travail Amazon Managed Service for Prometheus :

```
# CPU usage percentage
100 - (avg by (instance) (rate(node_cpu_seconds_total{mode="idle"}[5m])) * 100)

# Memory usage percentage
(1 - (node_memory_MemAvailable_bytes / node_memory_MemTotal_bytes)) * 100

# Request rate per second
sum(rate(http_requests_total[5m])) by (job)

# Error rate percentage
sum(rate(http_requests_total{status_code=~"5.."}[5m])) /
sum(rate(http_requests_total[5m])) * 100
```

```
# 95th percentile response time
histogram_quantile(0.95, sum(rate(http_request_duration_seconds_bucket[5m])) by (le))

# Top 5 instances by CPU usage
topk(5, avg by (instance) (cpu_usage))
```

Sécurisez vos requêtes métriques

Amazon Managed Service for Prometheus vous aide à sécuriser l'interrogation de vos métriques.

Utilisation AWS PrivateLink avec Amazon Managed Service pour Prometheus

Le trafic réseau pour interroger les métriques dans Amazon Managed Service for Prometheus peut être effectué via un point de terminaison Internet public ou par un point de terminaison VPC via. AWS PrivateLink Lorsque vous l'utilisez AWS PrivateLink, le trafic réseau provenant de vos VPC est sécurisé au sein du AWS réseau sans passer par l'Internet public. Pour créer un point de terminaison AWS PrivateLink VPC pour Amazon Managed Service for Prometheus, consultez. Utilisation d'Amazon Managed Service for Prometheus avec des points de terminaison de VPC d'interface

Authentification et autorisation

Gestion des identités et des accès AWS est un service Web qui vous permet de contrôler en toute sécurité l'accès aux AWS ressources. Vous pouvez utiliser IAM pour contrôler les personnes qui s'authentifient (sont connectées) et sont autorisées (disposent d'autorisations) à utiliser des ressources. Amazon Managed Service for Prometheus s'intègre à IAM pour vous aider à protéger vos données. Lorsque vous configurez Amazon Managed Service for Prometheus, vous devez créer des rôles IAM qui permettent aux serveurs Grafana d'interroger les métriques stockées dans les espaces de travail Amazon Managed Service for Prometheus. Pour plus d'informations sur IAM, consultez En quoi consiste IAM ?.

Une autre fonctionnalité AWS de sécurité qui peut vous aider à configurer Amazon Managed Service pour Prometheus est AWS le processus AWS de signature Signature Version 4 (SigV4). Signature Version 4 est le processus permettant d'ajouter des informations d'authentification aux AWS demandes envoyées par HTTP. Pour des raisons de sécurité, la plupart des demandes AWS

doivent être signées avec une clé d'accès, qui consiste en un identifiant de clé d'accès et une clé d'accès secrète. Ces deux clés sont généralement appelées informations d'identification de sécurité. Pour plus d'informations sur SigV4, consultez la section Processus de signature Signature Version 4.

Configuration d'Amazon Managed Grafana pour une utilisation avec Amazon Managed Service for Prometheus

Amazon Managed Grafana est un service entièrement géré pour Grafana open source qui simplifie la connexion à des éditeurs de logiciels indépendants indépendants et à des services permettant de visualiser AWS et d'analyser vos sources de données à grande échelle.

Amazon Managed Service for Prometheus prend en charge l'utilisation d'Amazon Managed Grafana pour interroger les métriques dans un espace de travail. Dans la console Amazon Managed Grafana, vous pouvez ajouter un espace de travail Amazon Managed Service for Prometheus en tant que source de données en découvrant vos comptes Amazon Managed Service for Prometheus existants. Amazon Managed Grafana gère la configuration des informations d'identification requises pour accéder à Amazon Managed Service for Prometheus. Pour obtenir des instructions détaillées sur la création d'une connexion à Amazon Managed Service for Prometheus à partir d'Amazon Managed Grafana, consultez les instructions du Guide de l'utilisateur Amazon Managed Grafana.

Vous pouvez également consulter vos alertes Amazon Managed Service for Prometheus dans Amazon Managed Grafana. Pour obtenir des instructions sur la configuration de l'intégration avec les alertes, consultez la section Intégrez des alertes à Amazon Managed Grafana ou à Grafana open source.

Connexion à Amazon Managed Grafana dans un VPC privé

Amazon Managed Service for Prometheus fournit un point de terminaison de service auquel Amazon Managed Grafana peut se connecter lors de l'interrogation des métriques et des alertes.

Vous pouvez configurer Amazon Managed Grafana pour utiliser un VPC privé (pour plus de détails sur la configuration d'un VPC privé dans Grafana, consultez la section Connecting to Amazon VPC dans le Guide de l'utilisateur Amazon Managed Grafana). Selon les paramètres, ce VPC peut ne pas avoir accès au point de terminaison de service Amazon Managed Service for Prometheus.

Pour ajouter Amazon Managed Service for Prometheus comme source de données à un espace de travail Amazon Managed Grafana configuré pour utiliser un VPC privé spécifique, vous devez d'abord connecter Amazon Managed Service for Prometheus au même VPC en créant un point

de terminaison de VPC. Pour plus d'informations sur la création d'un point de terminaison de VPC, consultez la section Création d'un point de terminaison de VPC d'interface pour Amazon Managed Service for Prometheus.

Configuration de Grafana open source ou Grafana Enterprise pour une utilisation avec Amazon Managed Service for Prometheus

Vous pouvez utiliser une instance de Grafana pour interroger vos statistiques dans Amazon Managed Service for Prometheus. Cette rubrique explique comment interroger les métriques d'Amazon Managed Service pour Prometheus à l'aide d'une instance autonome de Grafana.

Conditions préalables

Instance Grafana : vous devez disposer d'une instance Grafana capable de s'authentifier auprès d'Amazon Managed Service for Prometheus.

Amazon Managed Service for Prometheus prend en charge l'utilisation de Grafana version 7.3.5 et versions ultérieures pour l'interrogation des métriques dans un espace de travail. Les versions 7.3.5 et ultérieures incluent la prise en charge de l' AWS authentification Signature Version 4 (SigV4).

Pour vérifier votre version de Grafana, entrez la commande suivante, en la remplaçant par *grafana_install_directory* le chemin d'accès à votre installation de Grafana :

```
grafana_install_directory/bin/grafana-server -v
```

Si vous ne possédez pas encore de Grafana autonome ou si vous avez besoin d'une version plus récente, vous pouvez installer une nouvelle instance. Pour obtenir des instructions sur la configuration d'un Grafana autonome, voir Installer Grafana dans la documentation de Grafana. Pour plus d'informations sur la prise en main de Grafana, voir Getting started with Grafana dans la documentation de Grafana.

Compte AWS— Vous devez disposer des autorisations appropriées pour accéder à vos métriques Amazon Managed Service for Prometheus. Compte AWS

Pour configurer Grafana afin qu'il fonctionne avec Amazon Managed Service for Prometheus, vous devez être connecté à un compte disposant de la AmazonPrometheusQueryAccesspolitique ou des autorisations,, et. `aps:QueryMetrics` `aps:GetMetricMetadata` `aps:GetSeries` `aps:GetLabels` Pour de plus amples informations, veuillez consulter Autorisations et politiques IAM.

La section suivante décrit plus en détail la configuration de l'authentification depuis Grafana.

Étape 1 : Configurer AWS SigV4

Amazon Managed Service for Prometheus fonctionne Gestion des identités et des accès AWS avec (IAM) pour sécuriser tous les appels aux API Prometheus avec des informations d'identification IAM. Par défaut, la source de données Prometheus dans Grafana suppose que Prometheus ne nécessite aucune authentification. Pour permettre à Grafana de tirer parti des fonctionnalités d'authentification et d'autorisation d'Amazon Managed Service for Prometheus, vous devez activer la prise en charge de l'authentification SigV4 dans la source de données Grafana. Suivez les étapes indiquées sur cette page lorsque vous utilisez un serveur open source Grafana autogéré ou un serveur d'entreprise Grafana. Si vous utilisez Amazon Managed Grafana, l'authentification SigV4 est entièrement automatisée. Pour plus d'informations sur Amazon Managed Grafana, consultez la section What is Amazon Managed Grafana?.

Pour activer SigV4 sur Grafana, démarrez Grafana avec les variables d'environnement `AWS_SDK_LOAD_CONFIG` et `GF_AUTH_SIGV4_AUTH_ENABLED` définies sur `true`. La variable d'environnement `GF_AUTH_SIGV4_AUTH_ENABLED` remplace la configuration par défaut de Grafana pour activer le support SigV4. Pour de plus amples informations, veuillez consulter Configuration dans la documentation Grafana.

Linux

Pour activer SigV4 sur un serveur Grafana autonome sous Linux, entrez les commandes suivantes.

```
export AWS_SDK_LOAD_CONFIG=true
```

```
export GF_AUTH_SIGV4_AUTH_ENABLED=true
```

```
cd grafana_install_directory
```

```
./bin/grafana-server
```

Windows

Pour activer SigV4 sur un serveur Grafana autonome sous Windows à l'aide de l'invite de commande Windows, entrez les commandes suivantes.

```
set AWS_SDK_LOAD_CONFIG=true
```

```
set GF_AUTH_SIGV4_AUTH_ENABLED=true
```

```
cd grafana_install_directory
```

```
.\bin\grafana-server.exe
```

Étape 2 : Ajouter la source de données Prometheus dans Grafana

Les étapes suivantes expliquent comment configurer la source de données Prometheus dans Grafana pour interroger vos métriques Amazon Managed Service for Prometheus.

Pour ajouter la source de données Prometheus dans votre serveur Grafana

1. Ouvrez la console Grafana.
2. Sous Configurations, sélectionnez Sources de données.
3. Choisissez Add data source.
4. Choisissez Prometheus.
5. Pour l'URL HTTP, spécifiez le point de terminaison - l'URL de requête affiché sur la page de détails de l'espace de travail de la console Amazon Managed Service for Prometheus.
6. Dans l'URL HTTP que vous venez de spécifier, supprimez la chaîne `/api/v1/query` ajoutée à l'URL, car la source de données Prometheus l'ajoutera automatiquement.

   L'URL correcte doit ressembler à https://aps-workspaces.us-west-2.amazonaws.com/workspaces/ws-1234a5b6-78cd-901e-2fgh-3i45j6k178l9.
7. Sous Auth, sélectionnez le bouton SigV4 Auth pour l'activer.
8. Vous pouvez configurer l'autorisation SigV4 en spécifiant vos informations d'identification à long terme directement dans Grafana ou en utilisant une chaîne de fournisseurs par défaut. La spécification de vos informations d'identification à long terme vous permet de démarrer plus rapidement, et les étapes suivantes fournissent ces instructions en premier. Une fois que vous serez familiarisé avec l'utilisation de Grafana avec Amazon Managed Service for Prometheus, nous vous recommandons d'utiliser une chaîne de fournisseurs par défaut, qui offre plus de flexibilité et de sécurité. Pour plus d'informations sur la configuration de votre chaîne de fournisseurs par défaut, consultez la section Spécification des informations d'identification.

- Pour utiliser directement vos informations d'identification à long terme, procédez comme suit :

  a. Sous SigV4 Auth Details, choisissez Access & secret key pour le Fournisseur d'authentification.

  b. Pour Access Key ID, entrez votre ID de clé d'accès AWS .

  c. Pour Secret Access Key, entrez votre clé d'accès secrète AWS .

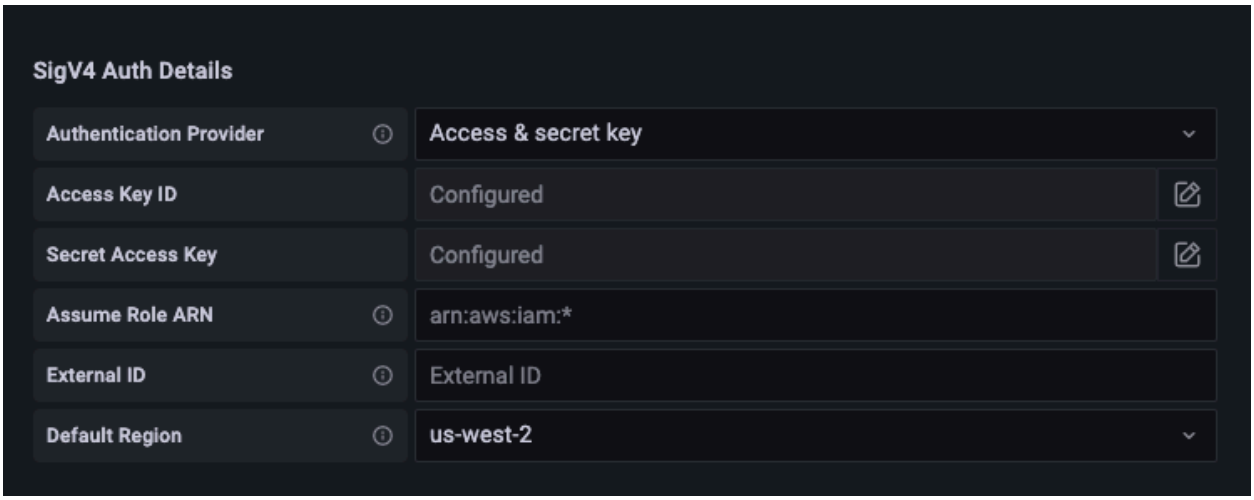
  d. Laissez les champs Assume Role ARN et External ID vides.

  e. Pour Default Region, choisissez la région de votre espace de travail Amazon Managed Service for Prometheus. Cette région doit correspondre à la région contenue dans l'URL répertoriée à l'étape 5.

  f. Choisissez Enregistrer et tester.

  Vous devez voir le message suivant : Data source is working.

  La capture d'écran suivante montre le paramètre détaillé d'authentification de la clé d'accès et de la clé secrète SigV4.



- Pour utiliser une chaîne de fournisseurs par défaut à la place (recommandée pour un environnement de production), procédez comme suit :

  a. Sous SigV4 Auth Details, choisissez AWS SDK Default pour le Fournisseur d'authentification.

  b. Laissez les champs Assume Role ARN et External ID vides.

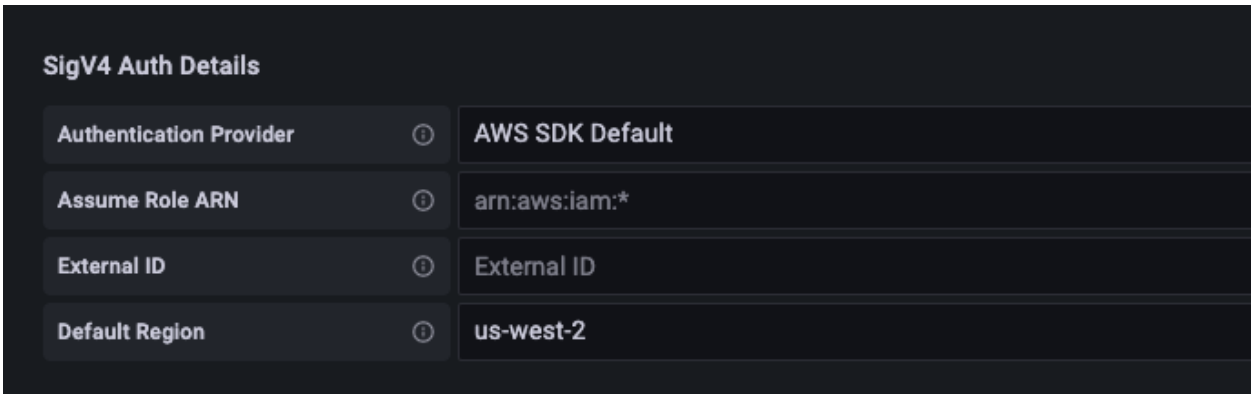
c. Pour Default Region, choisissez la région de votre espace de travail Amazon Managed Service for Prometheus. Cette région doit correspondre à la région contenue dans l'URL répertoriée à l'étape 5.

d. Choisissez Enregistrer et tester.

Vous devez voir le message suivant : Data source is working.

Si ce message ne s'affiche pas, la section suivante fournit des conseils de résolution des problèmes de connexion.

La capture d'écran suivante montre le paramètre détaillé d'authentification SigV4 par défaut du SDK.



9. Testez une requête ProMQL sur la nouvelle source de données :

a. Choisissez Explore.

b. Exécutez un exemple de requête ProMQL tel que :

```
prometheus_tsdb_head_series
```

Étape 3 : (facultatif) Résolution des problèmes si la sauvegarde du & test ne fonctionne pas

Dans la procédure précédente, si un message d'erreur s'affiche lorsque vous sélectionnez Enregistrer et tester, vérifiez les points suivants.

HTTP Error Not Found

Assurez-vous que l'ID de l'espace de travail indiqué dans l'URL est correct.

HTTP Error Forbidden

Cette erreur signifie que les informations d'identification ne sont pas valides. Vérifiez les éléments suivants :

- Vérifiez que la région spécifiée dans Default Region est correcte.
- Vérifiez que vos informations d'identification ne contiennent pas de fautes de frappe.
- Assurez-vous que les informations d'identification que vous utilisez sont conformes à la AmazonPrometheusQueryAccesspolitique. Pour de plus amples informations, veuillez consulter Autorisations et politiques IAM.
- Assurez-vous que les informations d'identification que vous utilisez ont accès à cet espace de travail Amazon Managed Service for Prometheus.

HTTP Error Bad Gateway

Consultez le journal du serveur Grafana pour résoudre cette erreur. Pour plus d'informations, consultez la section Dépannage dans la documentation Grafana.

Si vous le voyez**`Error http: proxy error: NoCredentialProviders: no valid providers in chain`**, la chaîne de fournisseurs d'informations d'identification par défaut n'a pas pu trouver d'informations d' AWS identification valide à utiliser. Assurez-vous d'avoir configuré vos informations d'identification conformément aux instructions de la section Spécification des informations d'identification. Si vous souhaitez utiliser une configuration partagée, assurez-vous que l'environnement `AWS_SDK_LOAD_CONFIG` est défini sur `true`.

Requête à l'aide de Grafana exécutée dans un cluster Amazon EKS

Amazon Managed Service for Prometheus prend en charge l'utilisation de Grafana version 7.3.5 et versions ultérieures pour l'interrogation des métriques dans un espace de travail Amazon Managed Service for Prometheus. Les versions 7.3.5 et ultérieures incluent la prise en charge de l' AWS authentification Signature Version 4 (SigV4).

Pour configurer Grafana afin qu'il fonctionne avec Amazon Managed Service for Prometheus, vous devez être connecté à un compte disposant de la AmazonPrometheusQueryAccesspolitique ou des autorisations,, et. `aps:QueryMetrics` `aps:GetMetricMetadata` `aps:GetSeries`

`aps:GetLabels` Pour de plus amples informations, veuillez consulter Autorisations et politiques IAM.

Configuration AWS SigV4

Grafana a ajouté une nouvelle fonctionnalité pour prendre en charge l'authentification AWS Signature Version 4 (SigV4). Pour plus d'informations, consultez Processus de signature Signature Version 4. Cette fonctionnalité est activée par défaut sur les serveurs Grafana. Les instructions suivantes pour activer cette fonctionnalité supposent que vous utilisez Helm pour déployer Grafana sur un cluster Kubernetes.

Pour activer SigV4 sur votre serveur Grafana 7.3.5 ou version ultérieure

1. Créez un nouveau fichier de mise à jour pour remplacer votre configuration Grafana et nommez-le `amp_query_override_values.yaml`.
2. Entrez le contenu qui suit dans le fichier, puis enregistrez le fichier. *`account-id`*Remplacez-le par l'identifiant du AWS compte sur lequel le serveur Grafana est exécuté.

```
serviceAccount:
    name: "amp-iamproxy-query-service-account"
    annotations:
        eks.amazonaws.com/role-arn: "arn:aws:iam::account-id:role/amp-iamproxy-
query-role"
grafana.ini:
  auth:
    sigv4_auth_enabled: true
```

   Dans le contenu de ce fichier YAML, `amp-iamproxy-query-role` correspond au nom du rôle que vous allez créer dans la section suivante, Configuration de rôles IAM pour les comptes de service. Vous pouvez remplacer ce rôle par votre propre nom de rôle si vous avez déjà créé un rôle pour interroger votre espace de travail.

Vous utiliserez ce fichier ultérieurement, dans Mise à niveau du serveur Grafana à l'aide de Helm.

Configuration de rôles IAM pour les comptes de service

Si vous utilisez un serveur Grafana dans un cluster Amazon EKS, nous vous recommandons d'utiliser des rôles IAM pour les comptes de service, également appelés rôles de service, pour votre contrôle d'accès. Lorsque vous le faites pour associer un rôle IAM à un compte de service Kubernetes, le

compte de service peut alors fournir des AWS autorisations aux conteneurs de n'importe quel pod utilisant ce compte de service. Pour plus d'informations, consultez la section Rôles IAM pour les comptes de service.

Si vous n'avez pas encore configuré ces rôles de service pour l'interrogation, suivez les instructions de la section Configuration de rôles IAM de comptes de service pour l'interrogation des métriques pour les configurer.

Vous devez ensuite ajouter le compte de service Grafana dans les conditions de la relation de confiance.

Pour ajouter le compte de service Grafana dans les conditions de la relation de confiance

1. À partir d'une fenêtre de terminal, déterminez l'espace de noms et le nom du compte de service de votre serveur Grafana. Par exemple, vous pouvez utiliser la commande suivante.

   ```
   kubectl get serviceaccounts -n grafana_namespace
   ```

2. Dans la console Amazon EKS, ouvrez le rôle IAM pour les comptes de service associés au cluster EKS.
3. Choisissez Modifier la relation d'approbation.
4. Mettez à jour la condition pour inclure l'espace de noms Grafana et le nom du compte de service Grafana obtenus dans le résultat de la commande à l'étape 1. Voici un exemple.

   JSON

   ```
   {
     "Version":"2012-10-17",
     "Statement": [
       {
         "Effect": "Allow",
         "Principal": {
           "Federated": "arn:aws:iam::111122223333:oidc-provider/oidc.eks.us-
   east-1.amazonaws.com/id/EXAMPLED539D4633E53DE1B71EXAMPLE"
         },
         "Action": "sts:AssumeRoleWithWebIdentity",
         "Condition": {
           "StringEquals": {
             "oidc.eks.us-east-1.amazonaws.com/
   id/EXAMPLED539D4633E53DE1B71EXAMPLE:sub": [
   ```

```
            "system:serviceaccount:aws-amp:amp-iamproxy-query-service-
account",
            "system:serviceaccount:grafana-namespace:grafana-service-account-
name"
          ],
          "oidc.eks.us-east-1.amazonaws.com/
id/EXAMPLED539D4633E53DE1B71EXAMPLE:aud": "sts.amazonaws.com"
        }
      }
    }
  ]
}
```

5. Choisissez Update Trust Policy (Mettre à jour la stratégie d'approbation).

Mise à niveau du serveur Grafana à l'aide de Helm

Cette étape met à niveau le serveur Grafana pour utiliser les entrées que vous avez ajoutées au fichier `amp_query_override_values.yaml` dans la section précédente.

Exécutez les commandes suivantes. Pour plus d'informations sur les Charts de Helm pour Grafana, consultez la section Grafana Community Kubernetes Helm Charts.

```
helm repo add grafana https://grafana.github.io/helm-charts
```

```
helm upgrade --install grafana grafana/grafana -n grafana_namespace -f ./
amp_query_override_values.yaml
```

Ajout de la source de données Prometheus dans Grafana

Les étapes suivantes expliquent comment configurer la source de données Prometheus dans Grafana pour interroger vos métriques Amazon Managed Service for Prometheus.

Pour ajouter la source de données Prometheus dans votre serveur Grafana

1. Ouvrez la console Grafana.
2. Sous Configurations, sélectionnez Sources de données.
3. Choisissez Add data source.
4. Choisissez Prometheus.

5. Pour l'URL HTTP, spécifiez le point de terminaison - l'URL de requête affiché sur la page de détails de l'espace de travail de la console Amazon Managed Service for Prometheus.
6. Dans l'URL HTTP que vous venez de spécifier, supprimez la chaîne `/api/v1/query` ajoutée à l'URL, car la source de données Prometheus l'ajoutera automatiquement.
7. Sous Auth, sélectionnez le bouton SigV4 Auth pour l'activer.

   Laissez les champs Assume Role ARN et External ID vides. Ensuite, pour Default Region, sélectionnez la région dans laquelle se trouve votre espace de travail Amazon Managed Service for Prometheus.
8. Choisissez Enregistrer et tester.

   Vous devez voir le message suivant : Data source is working.
9. Testez une requête ProMQL sur la nouvelle source de données :

   a. Choisissez Explore.

   b. Exécutez un exemple de requête ProMQL tel que :

   ```
   prometheus_tsdb_head_series
   ```

Requête à l'aide d' Prometheus-compatible API

Bien que l'utilisation d'un outil tel qu'Amazon Managed Grafana soit le moyen le plus simple de consulter et d'interroger vos statistiques, Amazon Managed Service for Prometheus prend également en charge plusieurs Prometheus-compatible API que vous pouvez utiliser pour interroger vos statistiques. Pour plus d'informations sur toutes les Prometheus-compatible API disponibles, consultezPrometheus-compatible APIs.

Les Prometheus-compatible API utilisent le langage de requête Prometheus, ProMQL, pour spécifier les données que vous souhaitez renvoyer. Pour plus de détails sur ProMQL et sa syntaxe, consultez la section Querying Prometheus dans la documentation de Prometheus.

Lorsque vous utilisez ces API pour interroger vos métriques, les demandes doivent être signées selon le processus de signature AWS Signature Version 4. Vous pouvez configurer AWS Signature Version 4 pour simplifier le processus de signature. Pour plus d'informations, consultez aws-sigv4-proxy.

La signature via le proxy AWS SigV4 peut être effectuée à l'aide `awscurl` de. La rubrique suivante Utilisation d'awscurl pour interroger les Prometheus-compatible API vous explique comment configurer `awscurl` SigV4. AWS

Rubriques

- Utilisez awscurl pour effectuer des requêtes avec des API Prometheus-compatible

Utilisez awscurl pour effectuer des requêtes avec des API Prometheus-compatible

Les demandes d'API pour Amazon Managed Service for Prometheus doivent être signées avec SigV4. Vous pouvez utiliser awscurl pour simplifier le processus d'interrogation.

Pour installer `awscurl`, Python 3 et le gestionnaire de package PIP doivent être installés.

Sur une instance Linux, la commande suivante installe `awscurl`.

```
$ pip3 install awscurl
```

Sur un ordinateur macOS, la commande suivante installe `awscurl`.

```
$ brew install awscurl
```

L'exemple suivant est un exemple de `awscurl` requête. Remplacez *`Region`* les *`QUERY`* entrées *`Workspace-id`* et par les valeurs appropriées à votre cas d'utilisation :

```
# Define the Prometheus query endpoint URL. This can be found in the Amazon Managed
 Service for Prometheus console page
# under the respective workspace.

$ export AMP_QUERY_ENDPOINT=https://aps-workspaces.Region.amazonaws.com/
workspaces/Workspace-id/api/v1/query

# credentials are infered from the default profile
$ awscurl -X POST --region Region \
                  --service aps "${AMP_QUERY_ENDPOINT}" -d 'query=QUERY'  --header
 'Content-Type: application/x-www-form-urlencoded'
```

> **Note**
>
> Votre chaîne de requête doit être codée en URL.

Pour une requête commequery=up, vous pouvez obtenir des résultats tels que :

```
{
  "status": "success",
  "data": {
    "resultType": "vector",
    "result": [
      {
        "metric": {
          "__name__": "up",
          "instance": "localhost:9090",
          "job": "prometheus",
          "monitor": "monitor"
        },
        "value": [
          1652452637.636,
          "1"
        ]
      },
    ]
  }
}
```

Pour qu'awscurl signe les requêtes fournies, vous devez transmettre les informations d'identification valides de l'une des manières suivantes :

- Fournissez l'ID de clé d'accès et la clé secrète du rôle IAM. Vous trouverez la clé d'accès et la clé secrète du rôle dans le https://console.aws.amazon.com/iam/.

  Par exemple :

```
$ export AMP_QUERY_ENDPOINT=https://aps-workspaces.Region.amazonaws.com/
workspaces/Workspace_id/api/v1/query

$ awscurl -X POST --region <Region> \
                  --access_key <ACCESS_KEY> \
                  --secret_key <SECRET_KEY> \
```

```
                  --service aps "$AMP_QUERY_ENDPOINT?query=<QUERY>"
```

- Référencez les fichiers de configuration stockés dans `.aws/credentials` et le fichier `/aws/config`. Vous pouvez également choisir de spécifier le nom du profil à utiliser. S'il n'est pas spécifié, le fichier `default` sera utilisé. Par exemple :

```
$ export AMP_QUERY_ENDPOINT=https://aps-workspaces.<Region>.amazonaws.com/workspaces/
<Workspace_ID>/api/v1/query
$ awscurl -X POST --region <Region> \
                  --profile <PROFILE_NAME>
                  --service aps "$AMP_QUERY_ENDPOINT?query=<QUERY>"
```

- Utilisez le profil d'instance associé à l'instance EC2.

Exécution de demandes de requête à l'aide du conteneur awscurl

Lorsqu'il n'est pas possible d'installer une version différente de Python et les dépendances associées, un conteneur peut être utilisé pour empaqueter l'application `awscurl` et ses dépendances. L'exemple suivant utilise un environnement d'exécution Docker pour le déploiement d'`awscurl`, mais tout environnement d'exécution et image conformes à l'OCI fonctionneront.

```
$ docker pull okigan/awscurl
$ export AMP_QUERY_ENDPOINT=https://aps-workspaces.Region.amazonaws.com/
workspaces/Workspace_id/api/v1/query
$ docker run --rm -it okigan/awscurl --access_key $AWS_ACCESS_KEY_ID  --secret_key
 $AWS_SECRET_ACCESS_KEY \ --region Region --service aps "$AMP_QUERY_ENDPOINT?
query=QUERY"
```

Obtenez des statistiques sur l'utilisation de vos requêtes pour chaque requête

La tarification des requêtes est basée sur le nombre total d'échantillons de requêtes traités en un mois à partir des requêtes exécutées. Vous pouvez obtenir des statistiques sur chaque requête que vous effectuez afin de suivre les échantillons traités. La réponse à une requête pour une `query` ou une `queryRange` API peut inclure les données statistiques relatives aux échantillons de requêtes traités en incluant le paramètre de requête `stats=all` dans la demande. Un `samples` objet est créé dans l'`stats`objet et les `stats` données sont renvoyées dans la réponse.

L'objet `samples` contient les attributs suivants :

Attribut	Description
`totalQueryableSamples`	Nombre total d'échantillons de requêtes traités. Il s'agit des informations à utiliser pour la facturation.
`totalQueryableSamplesPerStep`	Nombre d'échantillons de requêtes traités à chaque étape. Il s'agit d'un tableau de tableaux avec l'horodatage de l'époque et le nombre d'échantillons chargés à l'étape spécifique.

Les exemples de demandes et de réponses qui incluent les informations `stats` contenues dans la réponse sont les suivants :

Exemple pour `query` :

GET

```
endpoint/api/v1/query?query=up&time=1652382537&stats=all
```

Réponse

```
{
    "status": "success",
    "data": {
        "resultType": "vector",
        "result": [
            {
                "metric": {
                    "__name__": "up",
                    "instance": "localhost:9090",
                    "job": "prometheus"
                },
                "value": [
                    1652382537,
                    "1"
                ]
            }
        ],
```

```
        "stats": {
            "timings": {
                "evalTotalTime": 0.00453349,
                "resultSortTime": 0,
                "queryPreparationTime": 0.000019363,
                "innerEvalTime": 0.004508405,
                "execQueueTime": 0.000008786,
                "execTotalTime": 0.004554219
            },
            "samples": {
                "totalQueryableSamples": 1,
                "totalQueryableSamplesPerStep": [
                    [
                        1652382537,
                        1
                    ]
                ]
            }
        }
    }
}
```

Exemple pour `queryRange` :

GET

```
endpoint/api/v1/query_range?query=sum+%28rate+%28go_gc_duration_seconds_count%5B1m%5D
%29%29&start=1652382537&end=1652384705&step=1000&stats=all
```

Réponse

```
{
    "status": "success",
    "data": {
        "resultType": "matrix",
        "result": [
            {
                "metric": {},
                "values": [
                    [
                        1652383000,
                        "0"
                    ],
```

```
                    [
                        1652384000,
                        "0"
                    ]
                ]
            }
        ],
        "stats": {
            "samples": {
                "totalQueryableSamples": 8,
                "totalQueryableSamplesPerStep": [
                    [
                        1652382000,
                        0
                    ],
                    [
                        1652383000,
                        4
                    ],
                    [
                        1652384000,
                        4
                    ]
                ]
            }
        }
    }
}
```

Détection des anomalies

Amazon Managed Service for Prometheus fournit des fonctionnalités de détection des anomalies qui utilisent des algorithmes d'apprentissage automatique pour identifier automatiquement les modèles inhabituels dans vos données métriques. Cette fonctionnalité vous aide à détecter de manière proactive les problèmes potentiels, à réduire la fatigue liée aux alertes et à améliorer l'efficacité de votre surveillance en vous concentrant sur les comportements réellement anormaux plutôt que sur des seuils statiques.

La détection des anomalies dans Amazon Managed Service for Prometheus utilise l'algorithme Random Cut Forest (RCF), qui analyse les données de vos séries chronologiques pour établir des modèles de comportement normaux et identifier les écarts par rapport à ces modèles. L'algorithme s'adapte aux tendances saisonnières, gère les données manquantes avec élégance et fournit des scores de confiance pour les anomalies détectées.

Comment fonctionne la détection d'anomalies

La détection des anomalies par Amazon Managed Service for Prometheus utilise l'apprentissage automatique pour identifier des modèles inhabituels dans les données métriques sans configurer manuellement les seuils. Le système apprend les modèles de comportement normaux et les variations saisonnières, réduisant ainsi les faux positifs et permettant une détection précoce des problèmes. Il s'adapte en permanence aux modifications des applications, ce qui le rend adapté aux environnements cloud dynamiques.

La détection des anomalies surveille les indicateurs de performance des applications tels que les temps de réponse et les taux d'erreur, suit l'état de l'infrastructure en fonction de l'utilisation du processeur et de la mémoire, détecte les comportements inhabituels des utilisateurs, identifie les besoins en matière de planification des capacités grâce à l'analyse du trafic et surveille les indicateurs commerciaux pour détecter les changements inattendus. Cela fonctionne mieux avec des modèles prévisibles, des variations saisonnières ou des tendances de croissance graduelles.

L'algorithme Random Cut Forest (RCF) est utilisé pour analyser les données de séries chronologiques. RCF crée des arbres de décision qui partitionnent l'espace de données et identifient les points isolés éloignés de la distribution normale. L'algorithme apprend à partir des données entrantes pour créer un modèle dynamique du comportement normal pour chaque métrique.

Lorsqu'il est activé, il analyse les données historiques pour établir des modèles de référence et des tendances saisonnières, puis génère des prévisions pour les valeurs attendues et identifie les écarts. L'algorithme produit quatre sorties clés :

- upper_band - La limite supérieure des valeurs normales attendues
- lower_band - La limite inférieure des valeurs normales attendues
- score : score d'anomalie numérique indiquant à quel point le point de données est inhabituel
- valeur - La valeur métrique réelle observée

Mise en route avec la détection des anomalies

Pour commencer à utiliser la détection des anomalies avec vos métriques Prometheus, vous avez besoin de suffisamment de données historiques pour que l'algorithme apprenne les modèles normaux. Nous recommandons de disposer d'au moins 14 jours de données métriques cohérentes avant d'activer la détection des anomalies pour des résultats optimaux.

Vous pouvez prévisualiser le fonctionnement de la détection des anomalies avec vos métriques à l'aide de l'`PreviewAnomalyDetector`API. `PreviewAnomalyDetector`À utiliser pour tester l'algorithme par rapport à vos données historiques et évaluer son efficacité avant de l'implémenter dans le suivi de la production. Pour de plus amples informations, veuillez consulter PreviewAnomalyDetector API.

Lors de la mise en œuvre de la détection des anomalies, tenez compte des meilleures pratiques suivantes :

- Commencez par des indicateurs stables : commencez par des indicateurs présentant des modèles cohérents et évitez au départ des données très volatiles ou éparses.
- Utilisez des données agrégées : appliquez la détection des anomalies aux mesures agrégées (telles que les moyennes ou les sommes) plutôt qu'aux données brutes à cardinalité élevée pour améliorer les performances et la précision.
- Régler la sensibilité : ajustez les paramètres de l'algorithme en fonction de votre cas d'utilisation spécifique et de votre tolérance aux faux positifs par rapport aux anomalies manquées.
- Surveillez les performances de l'algorithme — Passez régulièrement en revue les anomalies détectées pour vous assurer que l'algorithme continue de fournir des informations précieuses au fur et à mesure de l'évolution de votre système.

PreviewAnomalyDetector API

Utilisez cette `PreviewAnomalyDetector` opération pour créer un point de terminaison qui montre comment vos données métriques seront analysées par l'algorithme de détection des anomalies au cours de la période spécifiée. Ce point de terminaison vous aide à évaluer et à valider les performances du détecteur avant sa mise en œuvre.

Verbes HTTP valides

`GET`, `POST`

Types de charge utile pris en charge

Paramètres codés en URL

`application/x-www-form-urlencoded` pour `POST`

Paramètres pris en charge

`query=<string>` Chaîne de requête d'expression Prometheus.

`start=<rfc3339 | unix_timestamp>` Démarrez l'horodatage si vous utilisez `query_range` pour effectuer des requêtes sur un intervalle de temps.

`end=<rfc3339 | unix_timestamp>` Arrêtez l'horodatage si vous utilisez `query_range` pour effectuer des requêtes sur un intervalle de temps.

`step=<duration | float>` Interrogez la durée de l'étape de résolution sous forme de `duration` ou sous forme d'un nombre `float` de secondes. À utiliser uniquement si vous utilisez `query_range` pour effectuer des requêtes sur un intervalle de temps, et nécessaire pour ce type de requêtes.

Formatage des paramètres de requête

Enveloppez votre expression ProMQL d'origine avec la pseudo-fonction RandomCutForest (RCF) dans le paramètre de requête. Pour plus d'informations, consultez le RandomCutForestConfigurationmanuel Amazon Managed Service for Prometheus API Reference.

La fonction RCF utilise le format suivant :

```
RCF(<query>
```

```
[,shingle size
[,sample size
[,ignore near expected from above
[,ignore near expected from below
[,ignore near expected from above ratio
[,ignore near expected from below ratio]]]]])
```

Tous les paramètres, à l'exception de la requête, sont facultatifs et utilisent des valeurs par défaut lorsqu'ils sont omis. La syntaxe minimale est la suivante :

```
RCF(<query>)
```

Vous devez encapsuler votre requête à l'aide d'une fonction d'agrégation. Pour utiliser des paramètres facultatifs spécifiques tout en omettant d'autres, laissez des positions vides dans la fonction :

```
RCF(<query>,,,,,1.0,1.0)
```

Cet exemple définit uniquement les paramètres de ratio qui ignorent les pics et les baisses de détection d'anomalies en fonction du rapport entre les valeurs attendues et observées.

Demande et réponse d'API

Les appels réussis renvoient le même format que l'QueryMetrics API. Outre les séries chronologiques d'origine, l'API renvoie ces nouvelles séries chronologiques lorsque suffisamment d'échantillons sont disponibles :

- `anomaly_detector_preview:lower_band`— Bande inférieure pour la valeur attendue du résultat de l'expression ProMQL
- `anomaly_detector_preview:score`— Score d'anomalie compris entre 0 et 1, où 1 indique un niveau de confiance élevé quant à une anomalie à ce point de données
- `anomaly_detector_preview:upper_band`— Bande supérieure pour la valeur attendue du résultat de l'expression ProMQL

Exemple de demande

```
POST /workspaces/workspace-id/anomalydetectors/preview
Content-Type: application/x-www-form-urlencoded

query=RCF%28avg%28vector%28time%28%29%29%29%2C%208%2C
%20256%29&start=1735689600&end=1735695000&step=1m
```

Exemple de réponse

```
200 OK
...

{
  "status": "success",
  "data": {
    "result": [
      {
        "metric": {},
        "values": [
          [
            1735689600,
            "1735689600"
          ],
          [
            1735689660,
            "1735689660"
          ],
          .........
        ]
      },
      {
        "metric": {
          "anomaly_detector_preview": "upper_band"
        },
        "values": [
          [
            1735693500,
            "1.7356943E9"
          ],
          [
            1735693560,
            "1.7356945E9"
          ]
          ],
```

```
          .........
        ]
      },
      {
        "metric": {
          "anomaly_detector_preview": "lower_band"
        },
        "values": [
          [
            1735693500,
            "1.7356928E9"
          ],
          [
            1735693560,
            "1.7356929E9"
          ],
          .........
        ]
      },
      {
        "metric": {
          "anomaly_detector_preview": "score"
        },
        "values": [
          [
            1735693500,
            "0.0"
          ],
          [
            1735695000,
            "0.0"
          ],
          .........
        ]
      }
    ],
    "resultType": "matrix"
  }
}
```

Utilisation de règles pour modifier ou surveiller les métriques au fur et à mesure de leur réception

Vous pouvez configurer des règles pour agir sur les métriques lorsqu'elles sont reçues par Amazon Managed Service for Prometheus. Ces règles peuvent surveiller les métriques ou même créer de nouvelles métriques calculées en fonction des métriques reçues.

Amazon Managed Service for Prometheus prend en charge deux types de règles qu'il évalue à intervalles réguliers :

- Les règles d'enregistrement permettent de précalculer des expressions fréquemment utilisées ou coûteuses en termes de calcul et d'enregistrer leurs résultats sous la forme d'un nouvel ensemble de séries temporelles. L'interrogation du résultat précalculé est souvent beaucoup plus rapide que l'exécution de l'expression d'origine chaque fois que cela est nécessaire.
- Les règles d'alerte permettent de définir des conditions d'alerte en fonction de ProMQL et d'un seuil. Lorsque la règle déclenche le seuil, une notification est envoyée au gestionnaire d'alertes, qui peut être configuré pour gérer les règles, ou pour les transmettre à une notification en aval aux destinataires tels qu'Amazon Simple Notification Service.

Pour utiliser les règles dans Amazon Managed Service for Prometheus, vous devez créer un ou plusieurs fichiers de règles YAML qui définissent les règles. Un fichier de règles Amazon Managed Service for Prometheus a le même format qu'un fichier de règles dans Prometheus autonome. Pour plus d'informations, consultez la section Defining Recording rules and Alerting rules dans la documentation Prometheus.

Un espace de travail peut avoir plusieurs fichiers de règles. Chaque fichier de règles distinct est contenu dans un espace de noms distinct. Le fait de disposer de plusieurs fichiers de règles vous permet d'importer des fichiers de règles Prometheus existants dans un espace de travail sans avoir à les modifier ou à les combiner. Les différents espaces de noms de groupes de règles peuvent également avoir des balises différentes.

Séquençage des règles

Dans un fichier de règles, les règles sont contenues dans des groupes de règles. Les règles d'un même groupe de règles dans un fichier de règles sont toujours évaluées de haut en bas. Par conséquent, dans les règles d'enregistrement, le résultat d'une règle d'enregistrement peut être utilisé dans le calcul d'une règle d'enregistrement ultérieure ou dans une règle d'alerte du même

groupe de règles. Toutefois, comme vous ne pouvez pas spécifier l'ordre dans lequel exécuter des fichiers de règles distincts, vous ne pouvez pas utiliser les résultats d'une règle d'enregistrement pour calculer une règle dans un autre groupe de règles ou un autre fichier de règles.

Rubriques

- Comprendre les autorisations IAM nécessaires à l'utilisation des règles
- Création d'un fichier de règles
- Chargez un fichier de configuration des règles sur Amazon Managed Service for Prometheus
- Modifier ou remplacer un fichier de configuration de règles
- Résoudre les problèmes liés aux évaluations des règles
- Dépannage des règles

Comprendre les autorisations IAM nécessaires à l'utilisation des règles

Vous devez autoriser les utilisateurs à utiliser des règles dans Amazon Managed Service for Prometheus. Créez une politique Gestion des identités et des accès AWS (IAM) avec les autorisations suivantes et attribuez-la à vos utilisateurs, groupes ou rôles.

> **Note**
>
> Pour plus d'informations sur IAM, consultez Gestion de l'identité et des accès dans Amazon Managed Service for Prometheus.

Politique d'accès aux règles d'utilisation

La politique suivante donne accès aux règles d'utilisation pour toutes les ressources de votre compte.

JSON

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

```
            "Action": [
                "aps:CreateRuleGroupsNamespace",
                "aps:ListRuleGroupsNamespaces",
                "aps:DescribeRuleGroupsNamespace",
                "aps:PutRuleGroupsNamespace",
                "aps:DeleteRuleGroupsNamespace"
            ],
            "Resource": "*"
        }
    ]
}
```

Politique d'accès à un seul espace de noms

Vous pouvez également créer des politiques d'accès à des politiques spécifiques. L'exemple de politique suivant donne accès uniquement à l'`RuleGroupNamespace` spécifié. Pour utiliser cette politique, remplacez*<account>*, *<region><workspace-id>*, et *<namespace-name>* par les valeurs appropriées pour votre compte.

Création d'un fichier de règles

Pour utiliser des règles dans Amazon Managed Service for Prometheus, vous devez créer un fichier de règles qui définit les règles. Un fichier de règles Amazon Managed Service for Prometheus est un fichier texte YAML dont le format est identique à celui d'un fichier de règles dans Prometheus autonome. Pour plus d'informations, consultez la section Définition des règles d'enregistrement et des règles d'alerte dans la documentation de Prometheus.

Voici un exemple de base de fichier de règles :

```
groups:
  - name: cpu_metrics
    interval: 60s
    rules:
     - record: avg_cpu_usage
       expr: avg(rate(node_cpu_seconds_total[5m])) by (instance)
     - alert: HighAverageCPU
       expr: avg_cpu_usage > 0.8
       for: 10m
       keep_firing_for: 20m
       labels:
```

```
        severity: critical
      annotations:
        summary: "Average CPU usage across cluster is too high"
```

Cet exemple crée un groupe de règles `cpu_metrics` qui est évalué toutes les 60 secondes. Ce groupe de règles crée une nouvelle métrique à l'aide d'une règle d'enregistrement, appelée `avg_cpu_usage` puis l'utilise dans une alerte. Voici une description de certaines des propriétés utilisées. Pour plus d'informations sur les règles d'alerte et les autres propriétés que vous pouvez inclure, consultez la section Règles d'alerte dans la documentation de Prometheus.

- `record: avg_cpu_usage`— Cette règle d'enregistrement crée une nouvelle métrique appeléeavg_cpu_usage.
- L'intervalle d'évaluation par défaut des groupes de règles est de 60 secondes si la `interval` propriété n'est pas spécifiée.
- `expr: avg(rate(node_cpu_seconds_total[5m])) by (instance)`— Cette expression de la règle d'enregistrement calcule le taux moyen d'utilisation du processeur au cours des 5 dernières minutes pour chaque nœud, en le regroupant par `instance` étiquette.
- `alert: HighAverageCPU`— Cette règle d'alerte crée une nouvelle alerte appelée `HighAverageCPU`
- `expr: avg_cpu_usage > 0.8` — Cette expression indique à l'alerte de rechercher des échantillons dans lesquels l'utilisation moyenne du processeur dépasse 80 %.
- `for: 10m`— L'alerte ne se déclenche que si l'utilisation moyenne du processeur dépasse 80 % pendant au moins 10 minutes.

  Dans ce cas, la métrique est calculée sous la forme d'une moyenne des 5 dernières minutes. L'alerte ne se déclenchera donc que s'il existe au moins deux échantillons consécutifs de 5 minutes (10 minutes au total) où l'utilisation moyenne du processeur est supérieure à 80 %.
- `keep_firing_for: 20m`— Cette alerte continuera à se déclencher jusqu'à ce que les échantillons soient inférieurs au seuil pendant au moins 20 minutes. Cela peut être utile pour éviter que l'alerte ne monte et ne baisse à plusieurs reprises.

> **Note**
>
> Vous pouvez créer un fichier de définition de règles localement, puis le télécharger sur Amazon Managed Service for Prometheus, ou vous pouvez créer, modifier et télécharger la définition directement dans la console Amazon Managed Service for Prometheus. Dans

tous les cas, les mêmes règles de mise en forme s'appliquent. Pour en savoir plus sur le chargement et la modification de votre fichier, consultezChargez un fichier de configuration des règles sur Amazon Managed Service for Prometheus.

Chargez un fichier de configuration des règles sur Amazon Managed Service for Prometheus

Une fois que vous savez quelles règles vous souhaitez inclure dans votre fichier de configuration des règles, vous pouvez soit le créer et le modifier dans la console, soit télécharger un fichier avec la console ou AWS CLI.

> **Note**
>
> Si vous utilisez un cluster Amazon EKS, vous pouvez également télécharger un fichier de configuration de règles à l'aide de AWS Controllers for Kubernetes.

Pour utiliser la console Amazon Managed Service for Prometheus afin de modifier ou de remplacer la configuration de vos règles et de créer l'espace de noms

1. Ouvrez la console Amazon Managed Service for Prometheus à l'adresse. https://console.aws.amazon.com/prometheus/
2. Dans le coin supérieur gauche de la page, cliquez sur l'icône du menu, puis sur Tous les espaces de travail.
3. Choisissez l'ID de l'espace de travail, puis cliquez sur l'onglet Rules management.
4. Choisissez Add namespace.
5. Choisissez Choose file, puis sélectionnez le fichier de définition des règles.

   Vous pouvez également créer et modifier un fichier de définition de règles directement dans la console Amazon Managed Service for Prometheus en sélectionnant Définir la configuration. Cela créera un exemple de fichier de définition par défaut que vous modifierez avant de le télécharger.
6. (Facultatif) Pour ajouter des balises à l'espace de noms, choisissez Ajouter une nouvelle balise.

   Ensuite, pour Key (Clé), saisissez un nom de balise Vous pouvez ajouter une valeur en option pour la balise dans Value (Valeur).

Pour ajouter une autre balise, choisissez Ajouter une nouvelle balise.

7. Sélectionnez Continuer. Amazon Managed Service for Prometheus crée un nouvel espace de noms portant le même nom que le fichier de règles que vous avez sélectionné.

Pour utiliser le plugin AWS CLI pour télécharger une configuration de gestionnaire d'alertes dans un espace de travail dans un nouvel espace de noms

1. Encodez en Base64 le contenu du fichier de votre gestionnaire d'alertes. Sous Linux, vous pouvez utiliser la commande suivante :

```
base64 input-file output-file
```

Sous macOS, vous pouvez utiliser la commande suivante :

```
openssl base64 input-file output-file
```

2. Entrez l'une des commandes suivantes pour créer l'espace de noms et télécharger le fichier.

Dans la AWS CLI version 2, entrez :

```
aws amp create-rule-groups-namespace --data file://path_to_base_64_output_file --
name namespace-name  --workspace-id my-workspace-id --region region
```

Dans la AWS CLI version 1, entrez :

```
aws amp create-rule-groups-namespace --data fileb://path_to_base_64_output_file --
name namespace-name  --workspace-id my-workspace-id --region region
```

3. Il faut compter quelques secondes pour que la configuration de votre gestionnaire d'alertes soit activée. Pour vérifier l'état, entrez la commande suivante :

```
aws amp describe-rule-groups-namespace --workspace-id workspace_id --
name namespace-name --region region
```

Si le `status` est `ACTIVE`, votre fichier de règles a pris effet.

Modifier ou remplacer un fichier de configuration de règles

Si vous souhaitez modifier les règles d'un fichier de règles que vous avez déjà chargé sur Amazon Managed Service for Prometheus, vous pouvez soit télécharger un nouveau fichier de règles pour remplacer la configuration existante, soit modifier la configuration actuelle directement dans la console. Vous pouvez éventuellement télécharger le fichier actuel, le modifier dans un éditeur de texte, puis télécharger la nouvelle version.

Pour utiliser la console Amazon Managed Service for Prometheus pour modifier la configuration de votre règles

1. Ouvrez la console Amazon Managed Service for Prometheus à l'adresse. https://console.aws.amazon.com/prometheus/
2. Dans le coin supérieur gauche de la page, cliquez sur l'icône du menu, puis sur Tous les espaces de travail.
3. Choisissez l'ID de l'espace de travail, puis cliquez sur l'onglet Rules management.
4. Sélectionnez le nom du fichier de configuration des règles que vous souhaitez modifier.
5. (Facultatif) Si vous souhaitez télécharger le fichier de configuration des règles actuel, choisissez Télécharger ou Copier.
6. Choisissez Modifier pour modifier la configuration directement dans la console. Choisissez Enregistrer lorsque vous avez terminé.

   Vous pouvez également choisir Remplacer la configuration pour télécharger un nouveau fichier de configuration. Dans ce cas, sélectionnez le nouveau fichier de définition des règles, puis choisissez Continuer pour le télécharger.

Pour utiliser le plugin AWS CLI pour modifier un fichier de configuration de règles

1. Encodez en Base64 le contenu de votre fichier de règles. Sous Linux, vous pouvez utiliser la commande suivante :

   ```
   base64 input-file output-file
   ```

   Sous macOS, vous pouvez utiliser la commande suivante :

   ```
   openssl base64 input-file output-file
   ```

2. Saisissez l'une des commandes suivantes pour télécharger le nouveau fichier.

   Dans la AWS CLI version 2, entrez :

   ```
   aws amp put-rule-groups-namespace --data file://path_to_base_64_output_file --
   name namespace-name  --workspace-id my-workspace-id --region region
   ```

   Dans la AWS CLI version 1, entrez :

   ```
   aws amp put-rule-groups-namespace --data fileb://path_to_base_64_output_file --
   name namespace-name  --workspace-id my-workspace-id --region region
   ```

3. Il faut compter quelques secondes pour que votre fichier de règles soit activé. Pour vérifier l'état, entrez la commande suivante :

   ```
   aws amp describe-rule-groups-namespace --workspace-id workspace_id --
   name namespace-name --region region
   ```

   Si le `status` est `ACTIVE`, votre fichier de règles a pris effet. En attendant, la version précédente de ce fichier de règles est toujours active.

Résoudre les problèmes liés aux évaluations des règles

Ce guide fournit des procédures de résolution étape par étape des problèmes courants liés à l'évaluation des règles dans Amazon Managed Service for Prometheus (AMP). Suivez ces procédures pour diagnostiquer et résoudre les problèmes liés à vos règles d'alerte et d'enregistrement.

Rubriques

- Valider l'état de déclenchement des alertes
- Résoudre les notifications d'alerte manquantes
- Vérifier l'état de santé des règles
- Utiliser le décalage dans les requêtes pour gérer les délais d'ingestion
- Problèmes courants et solutions correspondantes
- Bonnes pratiques pour l'évaluation des règles

Valider l'état de déclenchement des alertes

Lorsque vous résolvez des problèmes d'évaluation des règles, vérifiez d'abord si votre alerte s'est déclenchée en interrogeant la série `ALERTS` chronologique synthétique. Les séries `ALERTS` chronologiques incluent les libellés suivants :

- alertname — Le nom de l'alerte.
- alertstate — En attente ou en cours de lancement.
  - en attente — L'alerte est en attente pendant la durée spécifiée dans la `for` clause.
  - déclenchement — L'alerte a rempli les conditions pendant la durée spécifiée. Des libellés supplémentaires sont définis dans votre règle d'alerte.

> **Note**
>
> Lorsqu'une alerte est en cours de déclenchement ou en attente, la valeur d'échantillon est 1. Lorsque votre alerte est inactive, aucun échantillon n'est produit.

Résoudre les notifications d'alerte manquantes

Si des alertes se déclenchent mais que les notifications n'arrivent pas, vérifiez les paramètres d'Alertmanager suivants :

1. Vérifiez la configuration de votre Alertmanager : vérifiez que les itinéraires, les récepteurs et les paramètres sont correctement configurés. Passez en revue les paramètres de blocage des itinéraires, notamment les temps d'attente, les intervalles de temps et les étiquettes requises, qui peuvent affecter le déclenchement des alertes. Comparez les règles d'alerte avec les itinéraires et les récepteurs correspondants pour confirmer la bonne correspondance. Pour les itinéraires avec`time_interval`, vérifiez que les horodatages se situent dans les intervalles spécifiés.
2. Vérifiez les autorisations du destinataire des alertes : lorsque vous utilisez une rubrique Amazon SNS, vérifiez qu'AMP dispose des autorisations requises pour publier des notifications. Pour de plus amples informations, veuillez consulter Autoriser Amazon Managed Service for Prometheus à envoyer des messages d'alerte à votre rubrique Amazon SNS.
3. Validez la compatibilité de la charge utile du récepteur : vérifiez que votre récepteur d'alertes accepte le format de charge utile d'Alertmanager. Pour connaître les exigences relatives à Amazon SNS, consultez. Comprendre les règles de validation des messages Amazon SNS

4. Consultez les journaux d'Alertmanager — AMP fournit des journaux vendus par Alertmanager pour aider à résoudre les problèmes de notification. Pour de plus amples informations, veuillez consulter Surveillez les événements Prometheus via Amazon Managed Service grâce aux journaux CloudWatch.

Pour plus d'informations sur Alertmanager, consultez. Gestion et transfert des alertes dans Amazon Managed Service for Prometheus avec le gestionnaire d'alertes

Vérifier l'état de santé des règles

Des règles mal formées peuvent entraîner des échecs d'évaluation. Utilisez les méthodes suivantes pour identifier les raisons pour lesquelles une règle n'a pas pu être évaluée :

Example

Utiliser l' ListRules API

L'ListRulesAPI fournit des informations sur l'état des règles. Vérifiez les `lastError` champs `health` et pour diagnostiquer les problèmes.

Exemple de réponse :


```
{
  "status": "success",
  "data": {
    "groups": [
      {
        "name": "my_rule_group",
        "file": "my_namespace",
        "rules": [
          {
            "state": "firing",
            "name": "broken_alerting_rule",
            "query": "...",
            "duration": 0,
            "keepFiringFor": 0,
            "labels": {},
            "annotations": {},
            "alerts": [],
            "health": "err",
            "lastError": "vector contains metrics with the same labelset after applying
 alert labels",
```

```
            "type": "alerting",
            "lastEvaluation": "1970-01-01T00:00:00.00000000Z",
            "evaluationTime": 0.08
          }
        ]
      }
    ]
  }
}
```

Example

Utiliser des journaux vendus

L' ListRules API affiche uniquement les informations les plus récentes. Pour un historique plus détaillé, activez les journaux automatiques dans votre espace de travail pour accéder aux éléments suivants :

- Horodatage des échecs d'évaluation
- Messages d'erreur détaillés
- Données d'évaluation historiques

Exemple de message de journal vendu :

```
{
  "workspaceId": "ws-a2c55905-e0b4-4065-a310-d83ce597a391",
  "message": {
    "log": "Evaluating rule failed, name=broken_alerting_rule, group=my_rule_group,
 namespace=my_namespace, err=vector contains metrics with the same labelset after
 applying alert labels",
    "level": "ERROR",
    "name": "broken_alerting_rule",
    "group": "my_rule_group",
    "namespace": "my_namespace"
  },
  "component": "ruler"
}
```

Pour d'autres exemples de journaux provenant de Ruler ou d'Alertmanager, reportez-vous Dépannage des règles aux sections et. Gestion et transfert des alertes dans Amazon Managed Service for Prometheus avec le gestionnaire d'alertes

Utiliser le décalage dans les requêtes pour gérer les délais d'ingestion

Par défaut, les expressions sont évaluées sans décalage (requête instantanée), en utilisant les valeurs au moment de l'évaluation. Si l'ingestion des métriques est retardée, les règles d'enregistrement peuvent ne pas représenter les mêmes valeurs que lorsque vous évaluez manuellement l'expression après l'ingestion de toutes les métriques.

> **Tip**
>
> Vous pouvez configurer un décalage de requête de règle au niveau de l'espace de travail pour appliquer un délai global à toutes les évaluations de règles sans modifier les expressions de règle individuelles. Pour de plus amples informations, veuillez consulter Configurez votre espace de travail. Vous pouvez également utiliser le modificateur de décalage ProMQL sur des expressions individuelles. Pour plus d'informations, consultez la section Modificateur d'offset dans la documentation de Prometheus.

Exemple : gestion des métriques différées

Si votre règle est évaluée à 12 h, mais que le dernier échantillon pour la métrique date de 11 h 45 en raison d'un retard d'ingestion, la règle ne trouvera aucun échantillon à l'horodatage de 12 h 00. Pour atténuer ce problème, ajoutez un décalage, tel que :**`my_metric_name offset 15m`** .

Exemple : gestion des métriques provenant de plusieurs sources

Lorsque les métriques proviennent de sources différentes, telles que deux serveurs, elles peuvent être ingérées à des moments différents. Pour atténuer ce problème, créez une expression telle que : **`metric_from_server_A / metric_from_server_B`**

Si la règle évalue entre les temps d'ingestion du serveur A et du serveur B, vous obtiendrez des résultats inattendus. L'utilisation d'un décalage peut aider à aligner les temps d'évaluation.

Problèmes courants et solutions correspondantes

Lacunes dans l'enregistrement des données relatives aux règles

Si vous remarquez des lacunes dans les données de vos règles d'enregistrement par rapport à l'évaluation manuelle (lorsque vous exécutez directement l'expression ProMQL originale de la règle

d'enregistrement via l'API de requête ou l'interface utilisateur), cela peut être dû à l'une des raisons suivantes :

1. Délais d'évaluation longs : un groupe de règles ne peut pas faire l'objet de plusieurs évaluations simultanées. Si le temps d'évaluation dépasse l'intervalle configuré, les évaluations suivantes peuvent être manquées. Plusieurs évaluations manquées consécutives dépassant l'intervalle configuré peuvent entraîner l'obsolescence de la règle d'enregistrement. Pour plus d'informations, consultez Staleness dans la documentation de Prometheus. Vous pouvez surveiller la durée de l'évaluation `RuleGroupLastEvaluationDuration` à l'aide de la CloudWatch métrique pour identifier les groupes de règles dont l'évaluation prend trop de temps.
2. Surveillance des évaluations manquées : AMP fournit la `RuleGroupIterationsMissed` CloudWatch métrique permettant de suivre les cas où des évaluations sont ignorées. L' ListRules API affiche l'heure d'évaluation et l'heure de la dernière évaluation pour chacune rule/group, ce qui peut aider à identifier les modèles d'évaluations manquées. Pour de plus amples informations, veuillez consulter ListRules.

Recommandation : scindez les règles en groupes distincts

Pour réduire la durée des évaluations, divisez les règles en groupes de règles distincts. Les règles au sein d'un groupe s'exécutent de manière séquentielle, tandis que les groupes de règles peuvent s'exécuter en parallèle. Conservez les règles connexes qui dépendent les unes des autres dans le même groupe. En général, des groupes de règles plus restreints garantissent des évaluations plus cohérentes et réduisent les lacunes.

Bonnes pratiques pour l'évaluation des règles

1. Optimisez la taille des groupes de règles : limitez la taille des groupes de règles pour garantir des évaluations cohérentes. Regroupez les règles connexes, mais évitez les grands groupes de règles.
2. Définissez des intervalles d'évaluation appropriés : équilibre entre les alertes en temps opportun et la charge du système. Passez en revue les modèles de stabilité de vos indicateurs surveillés pour comprendre leurs plages de fluctuation normales.
3. Utilisez des modificateurs de décalage pour les mesures différées : ajoutez des décalages pour compenser les retards d'ingestion. Ajustez la durée du décalage en fonction des schémas d'ingestion observés.
4. Surveiller les performances d'évaluation — Suivez la `RuleGroupIterationsMissed` métrique. Consultez les délais d'évaluation dans l' ListRules API.

5. Valider les expressions de règles : assurez-vous que les expressions correspondent exactement entre les définitions de règles et les requêtes manuelles. Testez des expressions avec différentes plages de temps pour comprendre le comportement.
6. Vérifiez régulièrement l'état des règles : vérifiez l'absence d'erreurs dans les évaluations des règles. Surveillez les journaux vendus pour détecter les problèmes récurrents.

En suivant ces étapes de dépannage et les meilleures pratiques, vous pouvez identifier et résoudre les problèmes courants liés à l'évaluation des règles dans Amazon Managed Service for Prometheus.

Dépannage des règles

Surveillez les événements Prometheus via Amazon Managed Service grâce aux journaux CloudWatch vous permet de résoudre les problèmes liés au gestionnaire d'alertes et à l'outil de règle. Cette section contient des rubriques de dépannage relatives aux règles.

Lorsque le journal contient l'erreur d'échec de règles suivante


```
{
    "workspaceId": "ws-12345c67-89c0-4d12-345b-f14db70f7a99",
    "message": {
        "log": "Evaluating rule failed, name=failure,
 group=canary_long_running_vl_namespace, namespace=canary_long_running_vl_namespace,
 err=found duplicate series for the match group {dimension1=\\\"1\\\"} on the right
 hand-side of the operation: [{__name__=\\\"fake_metric2\\\", dimension1=\\\"1\\
\", dimension2=\\\"b\\\"}, {__name__=\\\"fake_metric2\\\", dimension1=\\\"1\\\",
 dimension2=\\\"a\\\"}];many-to-many matching not allowed: matching labels must be
 unique on one side",
        "level": "ERROR",
        "name": "failure",
        "group": "canary_long_running_vl_namespace",
        "namespace": "canary_long_running_vl_namespace"
    },
    "component": "ruler"
}
```


Cela signifie qu'une erreur s'est produite lors de l'exécution de la règle.

Action à exécuter

Utilisez le message d’erreur pour résoudre les problèmes d’exécution de règle.

Gestion et transfert des alertes dans Amazon Managed Service for Prometheus avec le gestionnaire d'alertes

Lorsque Amazon Managed Service for Prometheus déclenchent des règles d'alerte, le gestionnaire d'alertes gère les alertes envoyées. Il déduplique, regroupe et achemine les alertes vers les récepteurs en aval. Amazon Managed Service for Prometheus prend uniquement en charge Amazon Simple Notification Service en tant que récepteur et peut acheminer des messages vers les rubriques Amazon SNS sur le même compte. Vous pouvez également utiliser le gestionnaire d'alertes pour désactiver ou bloquer des alertes.

Le gestionnaire d'alertes fournit des fonctionnalités similaires à celles du gestionnaire d'alertes dans Prometheus.

Vous pouvez utiliser le fichier de configuration du gestionnaire d'alertes pour les opérations suivantes :

- Regroupement : le regroupement permet de rassembler des alertes similaires en une seule notification. Cela est particulièrement utile lors de pannes importantes lorsque de nombreux systèmes tombent en panne en même temps et que des centaines d'alertes peuvent se déclencher simultanément. Par exemple, supposons qu'une panne de réseau entraîne la défaillance simultanée de plusieurs nœuds. Si les alertes de ce type sont regroupées, le gestionnaire d'alertes ne vous envoie qu'une seule notification.

  Le regroupement des alertes et le calendrier des notifications groupées sont configurés par une arborescence de routage dans le fichier de configuration du gestionnaire d'alertes. Pour plus d'informations, consultez <route>.
- Inhibition : supprime les notifications pour certaines alertes si d'autres alertes sont déjà déclenchées. Par exemple, si une alerte indique qu'un cluster est inaccessible, vous pouvez configurer le gestionnaire d'alertes pour masquer toutes les autres alertes concernant ce cluster. Cela permet d'éviter les notifications pour des centaines ou des milliers d'alertes de déclenchement qui ne sont pas liées au problème réel. Pour plus d'informations sur la rédaction des règles d'inhibition, consultez <inhibit_rule>.
- Silences : masque les alertes pendant une durée spécifiée, par exemple pendant une période de maintenance. Les alertes entrantes sont vérifiées pour déterminer si elles correspondent à tous les critères d'égalité ou d'expression régulière d'un silence actif. Si c'est le cas, aucune notification n'est envoyée pour cette alerte.

Pour créer un silence, vous devez utiliser l'API `PutAlertManagerSilences`. Pour de plus amples informations, veuillez consulter PutAlertManagerSilences.

Modélisation de Prometheus

Prometheus autonome prend en charge la création de modèles à l'aide de fichiers modèles distincts. Les modèles peuvent notamment utiliser des conditions et formater des données.

Dans Amazon Managed Service for Prometheus, vous placez votre modèle dans le même fichier de configuration du gestionnaire d'alertes que celui de votre gestionnaire d'alertes.

Rubriques

- Comprendre les autorisations IAM nécessaires pour travailler avec le gestionnaire d'alertes
- Créez une configuration de gestionnaire d'alertes dans Amazon Managed Service pour Prometheus afin de gérer et d'acheminer les alertes
- Transférer les alertes à un récepteur d'alertes à l'aide du gestionnaire d'alertes dans Amazon Managed Service for Prometheus
- Téléchargez le fichier de configuration de votre gestionnaire d'alertes sur Amazon Managed Service for Prometheus
- Intégrez des alertes à Amazon Managed Grafana ou à Grafana open source
- Résoudre les problèmes liés au gestionnaire d'alertes avec les journaux CloudWatch

Comprendre les autorisations IAM nécessaires pour travailler avec le gestionnaire d'alertes

Vous devez autoriser les utilisateurs à utiliser le gestionnaire d'alertes dans Amazon Managed Service for Prometheus. Créez une politique Gestion des identités et des accès AWS (IAM) avec les autorisations suivantes et attribuez-la à vos utilisateurs, groupes ou rôles.

Créez une configuration de gestionnaire d'alertes dans Amazon Managed Service pour Prometheus afin de gérer et d'acheminer les alertes

Pour utiliser le gestionnaire d'alertes et le système de modélisation dans Amazon Managed Service for Prometheus, vous devez créer un fichier YAML de configuration de gestionnaire d'alertes. Un fichier de gestionnaire d'alertes Amazon Managed Service for Prometheus comporte deux sections principales :

- `template_files:` contient les modèles utilisés pour les messages envoyés par les destinataires. Pour plus d'informations, consultez les sections Référence de modèles et Exemples de modèles de la documentation Prometheus.
- `alertmanager_config:` contient la configuration du gestionnaire d'alertes. Il utilise la même structure qu'un fichier de configuration de gestionnaire d'alertes dans Prometheus autonome. Pour plus d'informations, consultez Configuration dans la documentation du gestionnaire d'alertes.

> **Note**
>
> La configuration `repeat_interval` décrite dans la documentation Prometheus ci-dessus comporte une limitation supplémentaire dans Amazon Managed Service for Prometheus. La valeur maximale autorisée est de cinq jours. Si vous définissez une durée supérieure à cinq jours, elle sera traitée comme cinq jours et les notifications seront à nouveau envoyées une fois la période de cinq jours écoulée.

> **Note**
>
> Vous pouvez également modifier le fichier de configuration directement dans la console Amazon Managed Service for Prometheus, mais il doit toujours respecter le format spécifié ici. Pour plus d'informations sur le téléchargement ou la modification d'un fichier de configuration, consultezTéléchargez le fichier de configuration de votre gestionnaire d'alertes sur Amazon Managed Service for Prometheus.

Dans Amazon Managed Service for Prometheus, le fichier de configuration de votre gestionnaire d'alertes doit contenir l'ensemble de la configuration dans une clé `alertmanager_config` située à la racine du fichier YAML.

Voici un exemple de fichier de configuration de gestionnaire d'alertes :

```
alertmanager_config: |
  route:
    receiver: 'default'
  receivers:
    - name: 'default'
      sns_configs:
      - topic_arn: arn:aws:sns:us-east-2:123456789012:My-Topic
        sigv4:
          region: us-east-2
        attributes:
          key: key1
          value: value1
```

Le seul récepteur actuellement pris en charge est Amazon Simple Notification Service (Amazon SNS). Si d'autres types de récepteurs sont répertoriés dans la configuration, ils seront rejetés.

Voici un autre exemple de fichier de configuration de gestionnaire d'alertes qui utilise à la fois le bloc `template_files` et le bloc `alertmanager_config`.

```
template_files:
  default_template: |
    {{ define "sns.default.subject" }}[{{ .Status | toUpper }}{{ if eq .Status
 "firing" }}:{{ .Alerts.Firing | len }}{{ end }}]{{ end }}
    {{ define "__alertmanager" }}AlertManager{{ end }}
    {{ define "__alertmanagerURL" }}{{ .ExternalURL }}/#/alerts?receiver={{ .Receiver |
 urlquery }}{{ end }}
alertmanager_config: |
  global:
  templates:
    - 'default_template'
  route:
    receiver: default
  receivers:
    - name: 'default'
      sns_configs:
      - topic_arn: arn:aws:sns:us-east-2:accountid:My-Topic
```

```
        sigv4:
          region: us-east-2
        attributes:
          key: severity
          value: SEV2
```

Bloc de modèle Amazon SNS par défaut

La configuration Amazon SNS par défaut utilise le modèle suivant, sauf si vous le remplacez explicitement.

```
{{ define "sns.default.message" }}{{ .CommonAnnotations.SortedPairs.Values | join "
 " }}
  {{ if gt (len .Alerts.Firing) 0 -}}
  Alerts Firing:
    {{ template "__text_alert_list" .Alerts.Firing }}
  {{- end }}
  {{ if gt (len .Alerts.Resolved) 0 -}}
  Alerts Resolved:
    {{ template "__text_alert_list" .Alerts.Resolved }}
  {{- end }}
{{- end }}
```

Transférer les alertes à un récepteur d'alertes à l'aide du gestionnaire d'alertes dans Amazon Managed Service for Prometheus

Lorsqu'une alerte est déclenchée par une règle d'alerte, elle est envoyée au gestionnaire d'alertes. Le gestionnaire d'alertes exécute des fonctions telles que la déduplication des alertes, l'inhibition des alertes pendant la maintenance ou leur regroupement selon les besoins. Il transmet ensuite l'alerte sous forme de message à un récepteur d'alerte. Vous pouvez configurer un récepteur d'alertes capable d'avertir les opérateurs, de recevoir des réponses automatisées ou de répondre aux alertes d'une autre manière.

Vous pouvez configurer Amazon Simple Notification Service (Amazon SNS) PagerDuty et en tant que destinataires d'alertes dans Amazon Managed Service for Prometheus. Les rubriques suivantes décrivent comment créer et configurer votre récepteur d'alertes.

Rubriques

- Utiliser Amazon SNS comme récepteur d'alertes
- Utiliser PagerDuty comme récepteur d'alertes

Utiliser Amazon SNS comme récepteur d'alertes

Vous pouvez utiliser une rubrique Amazon SNS existante comme récepteur d'alertes pour Amazon Managed Service for Prometheus, ou vous pouvez en créer une nouvelle. Nous vous recommandons d'utiliser une rubrique Standard, afin de pouvoir transférer des alertes de la rubrique vers un e-mail, un SMS ou le protocole HTTP.

Pour créer une rubrique Amazon SNS à utiliser comme récepteur de votre gestionnaire d'alertes, suivez les étapes de la section Étape 1 : Créer une rubrique. Assurez-vous de choisir Standard pour le type de rubrique.

Si vous souhaitez recevoir des e-mails chaque fois qu'un message est envoyé à cette rubrique Amazon SNS, suivez les étapes de la section Étape 2 : Créer un abonnement à la rubrique.

Que vous utilisiez une rubrique Amazon SNS nouvelle ou existante, vous aurez besoin du nom de ressource Amazon (ARN) de votre rubrique Amazon SNS pour effectuer les tâches suivantes.

Rubriques

- Autoriser Amazon Managed Service for Prometheus à envoyer des messages d'alerte à votre rubrique Amazon SNS
- Configurer le gestionnaire d'alertes pour envoyer des messages à votre rubrique Amazon SNS
- Configurer le gestionnaire d'alertes pour envoyer des messages à Amazon SNS au format JSON
- Configurer Amazon SNS pour envoyer des messages d'alerte vers d'autres destinations
- Comprendre les règles de validation des messages Amazon SNS

Autoriser Amazon Managed Service for Prometheus à envoyer des messages d'alerte à votre rubrique Amazon SNS

Vous devez autoriser Amazon Managed Service for Prometheus à envoyer des messages à votre rubrique Amazon SNS. La déclaration de politique suivante donnera cette autorisation. Il comprend une `Condition` déclaration visant à prévenir un problème de sécurité connu sous le nom de problème de confusion des adjoints. L'instruction `Condition` restreint l'accès à la rubrique Amazon

SNS pour autoriser uniquement les opérations provenant de ce compte spécifique et de l'espace de travail Amazon Managed Service for Prometheus. Pour de plus amples informations sur le problème de l'adjoint confus, veuillez consulter Prévention du cas de figure de l'adjoint désorienté entre services.

Pour autoriser Amazon Managed Service for Prometheus à envoyer des messages à votre rubrique Amazon SNS

1. Ouvrez la console Amazon SNS à l'adresse v3/home. https://console.aws.amazon.com/sns/
2. Dans le volet de navigation, choisissez Rubriques.
3. Choisissez le nom de la rubrique que vous utilisez avec Amazon Managed Service for Prometheus.
4. Choisissez Modifier.
5. Choisissez Stratégie d'accès et ajoutez l'instruction de stratégie suivante à la stratégie existante.

```
{
    "Sid": "Allow_Publish_Alarms",
    "Effect": "Allow",
    "Principal": {
        "Service": "aps.amazonaws.com"
    },
    "Action": [
        "sns:Publish",
        "sns:GetTopicAttributes"
    ],
    "Condition": {
        "ArnEquals": {
            "aws:SourceArn": "workspace_ARN"
        },
        "StringEquals": {
            "AWS:SourceAccount": "account_id"
        }
    },
    "Resource": "arn:aws:sns:region:account_id:topic_name"
}
```

[Facultatif] Si votre rubrique Amazon SNS est activée pour le chiffrement côté service (SSE), vous devez autoriser Amazon Managed Service for Prometheus à envoyer des messages à cette rubrique cryptée en ajoutant les `kms:Decrypt` autorisations `kms:GenerateDataKey*` et à la politique AWS KMS clé de la clé utilisée pour chiffrer la rubrique.

Par exemple, vous pouvez ajouter ce qui suit à la politique :

```
{
  "Statement": [{
    "Effect": "Allow",
    "Principal": {
      "Service": "aps.amazonaws.com"
    },
    "Action": [
      "kms:GenerateDataKey*",
      "kms:Decrypt"
    ],
    "Resource": "*"
  }]
}
```

Pour plus d'informations, veuillez consulter la section AWS Autorisations KMS pour SNS.

6. Sélectionnez Enregistrer les modifications.

> **Note**
>
> Par défaut, Amazon SNS crée la stratégie d'accès avec la condition `AWS:SourceOwner`. Pour plus d'informations, veuillez consulter la section SNS Access Policy.

> **Note**
>
> IAM suit la règle de la stratégie la plus restrictive en premier. Dans votre rubrique SNS, s'il existe un bloc de stratégie plus restrictif que le bloc de stratégie Amazon SNS documenté, l'autorisation pour la stratégie de la rubrique n'est pas accordée. Pour évaluer votre stratégie et savoir ce qui a été accordé, consultez la section Logique d'évaluation de stratégies.

Configuration des rubriques SNS pour les régions optionnelles

Vous pouvez l'utiliser `aps.amazonaws.com` pour configurer une rubrique Amazon SNS de la même manière Région AWS que votre espace de travail Amazon Managed Service for Prometheus. Pour utiliser un sujet SNS provenant d'une non-opt-in région (telle que us-east-1) avec une région

optionnelle (telle que af-south-1), vous devez utiliser le format principal du service régional. Dans le principe du service régional, remplacez *us-east-1* par la non-opt-in région que vous souhaitez utiliser :**aps.*us-east-1*.amazonaws.com**.

Le tableau suivant répertorie les régions optionnelles et les principaux services régionaux correspondants :

Régions adhérentes et leurs principaux de services régionaux

Nom de la région	Région	Directeur du service régional
Afrique (Le Cap)	af-south-1	af-south-1.aps.amazonaws.com
Asie-Pacifique (Hong Kong)	ap-east-1	ap-east-1.aps.amazonaws.com
Asie-Pacifique (Thaïlande)	ap-southeast-7	ap-southeast-7.aps.amazonaws.com
Europe (Milan)	eu-south-1	eu-south-1.aps.amazonaws.com
Europe (Zurich)	eu-central-2	eu-central-2.aps.amazonaws.com
Moyen-Orient (EAU)	me-central-1	me-central-1.aps.amazonaws.com
Asie-Pacifique (Malaisie)	ap-southeast-5	ap-southeast-5.aps.amazonaws.com

Pour plus d'informations sur l'activation d'une région optionnelle, consultez la section Gestion Régions AWS du guide de l'utilisateur IAM dans le. Référence générale d'Amazon Web Services

Lorsque vous configurez votre rubrique Amazon SNS pour ces régions optionnelles, assurez-vous d'utiliser le principal de service régional approprié pour permettre la diffusion des alertes entre les régions.

Prévention du cas de figure de l'adjoint désorienté entre services

Le problème de député confus est un problème de sécurité dans lequel une entité qui n'est pas autorisée à effectuer une action peut contraindre une entité plus privilégiée à le faire. En AWS, l'usurpation d'identité interservices peut entraîner la confusion des adjoints. L'usurpation d'identité entre services peut se produire lorsqu'un service (le service appelant) appelle un autre service (le service appelé). Le service appelant peut être manipulé et ses autorisations utilisées pour agir sur les ressources d'un autre client auxquelles on ne serait pas autorisé à accéder autrement. Pour éviter cela, AWS fournit des outils qui vous aident à protéger vos données pour tous les services avec des principaux de service qui ont eu accès aux ressources de votre compte.

Nous vous recommandons d'utiliser les clés de contexte de condition globale `aws:SourceArn` et `aws:SourceAccount` dans les politiques de ressources afin de limiter les autorisations accordées à la ressource par Amazon Managed Service for Promotheus. Si vous utilisez les deux clés de contexte de condition globale, la valeur `aws:SourceAccount` et le compte de la valeur `aws:SourceArn` doit utiliser le même ID de compte lorsqu'il est utilisé dans la même déclaration de stratégie.

La valeur de `aws:SourceArn` doit être l'ARN de l'espace de travail Amazon Managed Service for Prometheus.

Le moyen le plus efficace de se protéger contre le problème de député confus consiste à utiliser la clé de contexte de condition globale `aws:SourceArn` avec l'ARN complet de la ressource. Si vous ne connaissez pas l'ARN complet de la ressource ou si vous spécifiez plusieurs ressources, utilisez la clé de contexte de condition globale `aws:SourceArn` avec des caractères génériques (*) pour les parties inconnues de l'ARN. Par exemple, `arn:aws:`*`servicename`*`::`*`123456789012`*`:*`.

La stratégie présentée à la section Autoriser Amazon Managed Service for Prometheus à envoyer des messages d'alerte à votre rubrique Amazon SNS montre comment utiliser les clés de contexte de condition globale `aws:SourceArn` et `aws:SourceAccount` dans Amazon Managed Service for Prometheus afin d'éviter le problème de l'adjoint confus.

Configurer le gestionnaire d'alertes pour envoyer des messages à votre rubrique Amazon SNS

Une fois que vous avez créé une rubrique Amazon SNS de type standard (nouvelle ou existante), vous pouvez l'ajouter à la configuration de votre gestionnaire d'alertes en tant que récepteur d'alertes. Le gestionnaire d'alertes peut transmettre vos alertes à un récepteur d'alertes configuré. Pour terminer, vous devez connaître le nom de ressource Amazon (ARN) de votre rubrique Amazon SNS.

Pour plus d'informations sur la configuration du récepteur Amazon SNS, consultez la section <sns_configs> dans la documentation de configuration de Prometheus.

Propriétés non prises en charge

Amazon Managed Service for Prometheus prend en charge Amazon SNS en tant que récepteur d'alertes. Cependant, en raison de contraintes de service, toutes les propriétés du récepteur Amazon SNS ne sont pas prises en charge. Les propriétés suivantes ne sont pas autorisées dans un fichier de configuration de gestionnaire d'alertes Amazon Managed Service for Prometheus :

- `api_url:` : Amazon Managed Service for Prometheus définit `api_url` pour vous. Cette propriété n'est donc pas autorisée.
- `Http_config` : cette propriété vous permet de définir des proxys externes. Amazon Managed Service for Prometheus ne prend actuellement pas en charge cette fonctionnalité.

En outre, les paramètres SigV4 doivent avoir une propriété Région. Sans la propriété Région, Amazon Managed Service for Prometheus ne dispose pas de suffisamment d'informations pour effectuer la demande d'autorisation.

Pour configurer le gestionnaire d'alertes avec votre rubrique Amazon SNS comme destinataire

1. Si vous utilisez un fichier de configuration de gestionnaire d'alertes existant, ouvrez-le dans un éditeur de texte.
2. S'il existe actuellement des récepteurs autres qu'Amazon SNS dans le bloc `receivers`, supprimez-les. Vous pouvez configurer plusieurs rubriques Amazon SNS pour qu'elles soient des récepteurs en les plaçant dans des blocs `sns_config` distincts au sein du bloc `receivers`.
3. Ajoutez le bloc YAML suivant dans la section `receivers`.

```
- name: name_of_receiver
  sns_configs:
    - sigv4:
        region: Région AWS
      topic_arn: ARN_of_SNS_topic
      subject: yoursubject
      attributes:
        key: yourkey
        value: yourvalue
```

Si aucun `subject` n'est spécifié, par défaut, un objet est généré avec le modèle par défaut avec le nom et les valeurs de l'étiquette, ce qui peut entraîner une valeur trop longue pour SNS. Pour modifier le modèle appliqué à l'objet, reportez-vous à la section Configurer le gestionnaire d'alertes pour envoyer des messages à Amazon SNS au format JSON du présent guide.

Vous devez à présent télécharger le fichier de configuration de votre gestionnaire d'alertes sur Amazon Managed Service for Prometheus. Pour de plus amples informations, veuillez consulter Téléchargez le fichier de configuration de votre gestionnaire d'alertes sur Amazon Managed Service for Prometheus.

Configurer le gestionnaire d'alertes pour envoyer des messages à Amazon SNS au format JSON

Par défaut, le gestionnaire d'alertes Amazon Managed Service for Prometheus produit des messages sous forme de liste en texte brut. Cela peut être plus difficile à analyser pour les autres services. Vous pouvez configurer le gestionnaire d'alertes pour qu'il envoie des alertes au format JSON à la place. Le JSON peut simplifier le traitement des messages en aval d'Amazon SNS dans AWS Lambda ou dans les points de terminaison recevant des webhooks. Au lieu d'utiliser le modèle par défaut, vous pouvez définir un modèle personnalisé pour afficher le contenu du message au format JSON, ce qui facilite l'analyse dans les fonctions en aval.

Pour envoyer des messages du gestionnaire d'alertes à Amazon SNS au format JSON, mettez à jour la configuration de votre gestionnaire d'alertes afin qu'il contienne le code suivant dans votre section racine `template_files` :

```
default_template: |
   {{ define "sns.default.message" }}{{ "{" }}"receiver": "{{ .Receiver }}","status":
 "{{ .Status }}","alerts": [{{ range $alertIndex, $alerts := .Alerts }}{{ if
 $alertIndex }}, {{ end }}{{ "{" }}"status": "{{ $alerts.Status }}"{{ if
 gt (len $alerts.Labels.SortedPairs) 0 -}},"labels": {{ "{" }}{{ range
 $index, $label := $alerts.Labels.SortedPairs }}{{ if $index }},
 {{ end }}"{{ $label.Name }}": "{{ $label.Value }}"{{ end }}
{{ "}" }}{{- end }}{{ if gt (len $alerts.Annotations.SortedPairs )
 0 -}},"annotations": {{ "{" }}{{ range $index, $annotations :=
 $alerts.Annotations.SortedPairs }}{{ if $index }}, {{ end }}"{{ $annotations.Name }}":
 "{{ $annotations.Value }}"{{ end }}{{ "}" }}{{- end }},"startsAt":
 "{{ $alerts.StartsAt }}","endsAt": "{{ $alerts.EndsAt }}","generatorURL":
 "{{ $alerts.GeneratorURL }}","fingerprint": "{{ $alerts.Fingerprint }}"{{ "}" }}
{{ end }}]{{ if gt (len .GroupLabels) 0 -}},"groupLabels": {{ "{" }}{{ range
 $index, $groupLabels := .GroupLabels.SortedPairs }}{{ if $index }},
```

```
 {{ end }}"{{ $groupLabels.Name }}": "{{ $groupLabels.Value }}"{{ end }}
{{ "}" }}{{- end }}{{ if gt (len .CommonLabels) 0 -}},"commonLabels": {{ "{" }}
{{ range $index, $commonLabels := .CommonLabels.SortedPairs }}{{ if $index }},
 {{ end }}"{{ $commonLabels.Name }}": "{{ $commonLabels.Value }}"{{ end }}{{ "}" }}{{-
 end }}{{ if gt (len .CommonAnnotations) 0 -}},"commonAnnotations": {{ "{" }}{{ range
 $index, $commonAnnotations := .CommonAnnotations.SortedPairs }}{{ if $index }},
 {{ end }}"{{ $commonAnnotations.Name }}": "{{ $commonAnnotations.Value }}"{{ end }}
{{ "}" }}{{- end }}{{ "}" }}{{ end }}
   {{ define "sns.default.subject" }}[{{ .Status | toUpper }}{{ if eq .Status
 "firing" }}:{{ .Alerts.Firing | len }}{{ end }}]{{ end }}
```

> **Note**
>
> Ce modèle crée des données au format JSON à partir de données alphanumériques. Si vos données comportent des caractères spéciaux, encodez-les avant d'utiliser ce modèle.

Pour vous assurer que ce modèle est utilisé dans les notifications sortantes, référencez-le dans votre bloc `alertmanager_config` comme suit :

```
alertmanager_config: |
  global:
  templates:
    - 'default_template'
```

> **Note**
>
> Ce modèle est destiné à l'ensemble du corps du message au format JSON. Ce modèle remplace l'intégralité du corps du message. Vous ne pouvez pas remplacer le corps du message si vous souhaitez utiliser ce modèle spécifique. Tout remplacement effectué manuellement a la priorité sur le modèle.

Pour plus d'informations sur :

- Le fichier de configuration du gestionnaire d'alertes, voir Créez une configuration de gestionnaire d'alertes dans Amazon Managed Service pour Prometheus afin de gérer et d'acheminer les alertes.
- Le chargement de votre fichier de configuration, voir Téléchargez le fichier de configuration de votre gestionnaire d'alertes sur Amazon Managed Service for Prometheus.

Configurer Amazon SNS pour envoyer des messages d'alerte vers d'autres destinations

Amazon Managed Service for Prometheus peut uniquement envoyer des messages d'alerte à Amazon Simple Notification Service (Amazon SNS). Pour envoyer ces messages vers d'autres destinations, telles que le courrier électronique, le webhook, Slack, etc. OpsGenie, vous devez configurer Amazon SNS pour transférer les messages vers ces points de terminaison.

Les sections suivantes décrivent la configuration d'Amazon SNS pour transférer les alertes vers d'autres destinations.

Rubriques

- E-mail
- Webhook
- Slack
- OpsGenie

E-mail

Pour configurer une rubrique Amazon SNS pour envoyer des messages par e-mail, créez un abonnement. Dans la console Amazon SNS, choisissez l'onglet Abonnements pour ouvrir la page de liste Abonnements. Choisissez Créer un abonnement, puis sélectionnez E-mail. Amazon SNS envoie un e-mail de confirmation à l'adresse e-mail répertoriée. Après avoir accepté la confirmation, vous pouvez recevoir des notifications Amazon SNS sous forme d'e-mails provenant de la rubrique à laquelle vous vous êtes abonné. Pour plus d'informations, consultez Abonnement à une rubrique Amazon SNS.

Webhook

Pour configurer une rubrique Amazon SNS pour envoyer des messages à un point de terminaison webhook, créez un abonnement. Dans la console Amazon SNS, choisissez l'onglet Abonnements pour ouvrir la page de liste Abonnements. Choisissez Créer un abonnement, puis sélectionnez HTTP/HTTPS. Après avoir créé l'abonnement, vous devez suivre les étapes de confirmation pour l'activer. Lorsqu'il est actif, votre point de terminaison HTTP doit recevoir les notifications Amazon SNS. Pour plus d'informations, consultez Abonnement à une rubrique Amazon SNS. Pour plus d'informations sur l'utilisation de webhooks Slack pour publier des messages vers différentes destinations, consultez Comment utiliser les webhooks pour publier des messages Amazon SNS sur Amazon Chime, Slack ou Microsoft Teams ?

Slack

Pour configurer une rubrique Amazon SNS afin qu'elle envoie des messages à Slack, deux options s'offrent à vous. Vous pouvez soit intégrer email-to-channel l'intégration de Slack, qui permet à Slack d'accepter des e-mails et de les transférer vers un canal Slack, soit utiliser une fonction Lambda pour réécrire la notification Amazon SNS envoyée à Slack. Pour plus d'informations sur le transfert d'e-mails vers les chaînes Slack, consultez Confirmation de l'abonnement aux AWS rubriques SNS pour Slack Webhook. Pour plus d'informations sur la construction d'une fonction Lambda pour convertir les messages Amazon SNS en messages Slack, consultez la section How to integrate Amazon Managed Service for Prometheus with Slack.

OpsGenie

Pour plus d'informations sur la configuration d'une rubrique Amazon SNS vers laquelle envoyer des messages OpsGenie, consultez Intégrer Opsgenie à Amazon SNS entrant.

Comprendre les règles de validation des messages Amazon SNS

Amazon Simple Notification Service (Amazon SNS) exige que les messages répondent à certaines normes. Les messages qui ne répondent pas à ces normes seront modifiés dès leur réception. Les messages d'alerte seront validés, tronqués ou modifiés, si nécessaire, par le récepteur Amazon SNS selon les règles suivantes :

- Le message contient des caractères non UTF.
  - Le message sera remplacé par Error - il ne s'agit pas d'une chaîne codée en UTF-8 valide.
  - Un attribut de message sera ajouté avec la clé tronquée et la valeur true.
  - Un attribut de message sera ajouté avec la clé Modified et la valeur Message : Error - chaîne codée en UTF-8 non valide.
- Le message est vide.
  - Le message sera remplacé par Erreur - Le message ne doit pas être vide.
  - Un attribut de message sera ajouté avec la clé « modifié » et la valeur « Message : Error - Le message ne doit pas être vide ».
- Le message a été tronqué.
  - Le contenu du message sera tronqué.
  - Un attribut de message sera ajouté avec la clé tronquée et la valeur true.
  - Un attribut de message sera ajouté avec la clé « modifié » et la valeur Message : Erreur - Le message a été tronqué de ***X*** Ko, car il dépasse la limite de 256 Ko.

- Le sujet contient des caractères de contrôle ou non ASCII.
  - Si le sujet contient des caractères de contrôle ou des caractères non ASCII, SNS remplace le sujet par Error (contient des caractères de contrôle ou non ASCII).
  - Pour les sujets des e-mails SNS, supprimez les caractères de contrôle, tels que les nouvelles lignes :. \n
- L'objet n'est pas au format ASCII.
  - Le sujet sera remplacé par Erreur - contient des caractères ASCII non imprimables.
  - Un attribut de message sera ajouté avec la clé Modified et la valeur Subject : Error - contient des caractères ASCII non imprimables.
- L'objet a été tronqué.
  - Le contenu de l'objet sera tronqué.
  - Un attribut de message sera ajouté avec la clé « modifié » et la valeur « Objet : erreur » - Le nombre de ***X*** caractères de l'objet a été tronqué, car il dépasse la limite de 100 caractères.
- La clé/valeur de l'attribut de message n'est pas valide.
  - L'attribut de message non valide sera supprimé.
  - Un attribut de message sera ajouté avec la clé de modification et la valeur MessageAttribute: Erreur - si les attributs ***X*** du message ont été supprimés car non valides MessageAttributeKey ou MessageAttributeValue.
- L'attribut de message a été tronqué.
  - Les attributs de message supplémentaires seront supprimés.
  - Un attribut de message sera ajouté avec la clé « modifié » et la valeur « Erreur » MessageAttribute : si les attributs ***X*** du message ont été supprimés, car il dépasse la limite de 256 Ko.

Utiliser PagerDuty comme récepteur d'alertes

Vous pouvez configurer Amazon Managed Service for Prometheus pour qu'il envoie des alertes directement à. PagerDuty Cette intégration nécessite que vous stockiez votre clé PagerDuty d'intégration AWS Secrets Manager et que vous autorisiez Amazon Managed Service for Prometheus à lire le secret.

PagerDuty l'intégration permet d'automatiser les flux de travail de réponse aux incidents et de garantir que les alertes critiques parviennent aux bons membres de l'équipe au bon moment. Lorsque vous l'utilisez PagerDuty en tant que récepteur d'alertes, vous pouvez tirer parti des politiques d'escalade,

PagerDuty de la planification des appels et des fonctionnalités de gestion des incidents pour garantir que les alertes sont reconnues et résolues rapidement. Cette intégration est particulièrement utile pour les environnements de production où une réponse rapide aux problèmes du système est essentielle pour maintenir la disponibilité des services et respecter les exigences des SLA. Pour plus d'informations, consultez la base de PagerDuty connaissances sur le PagerDuty site Web.

PagerDuty options de configuration

Option	Description	Obligatoire
`routing_key`	La clé PagerDuty de routage pour une intégration sur un service. Vous devez le spécifier en tant qu'ARN du Gestionnaire de Secrets	Oui
`service_key`	La clé PagerDuty de service pour une intégration sur un service. Vous devez le spécifier en tant qu'ARN du Gestionnaire de Secrets	Oui (pour l'API Events v1)
`client`	L'identification du client du notifiant	Non
`client_url`	Un lien retour vers l'expéditeur de la notification	Non
`description`	Description de l'incident	Non
`details`	Un ensemble de key/value paires arbitraires qui fournissent plus de détails sur l'incident	Non
`severity`	Gravité de l'incident	Non
`class`	La classe, ou le type, de l'événement	Non

Option	Description	Obligatoire
`component`	Composant de la machine source responsable de l'événement	Non
`group`	Regroupement logique de composants	Non
`source`	L'emplacement unique du système concerné	Non

> **Note**
>
> Les `http_config` option`url`, `service_key_file``routing_key_file`, et ne sont pas prises en charge.

Les rubriques suivantes décrivent comment procéder à la configuration PagerDuty en tant que récepteur d'alertes dans Amazon Managed Service for Prometheus.

Rubriques

- Configuration AWS Secrets Manager et autorisations
- Configurer le gestionnaire d'alertes pour envoyer des alertes à PagerDuty

Configuration AWS Secrets Manager et autorisations

Avant de pouvoir envoyer des alertes à PagerDuty, vous devez stocker en toute sécurité votre clé PagerDuty d'intégration et configurer les autorisations nécessaires. Ce processus implique de créer un secret AWS Secrets Manager, de le chiffrer avec une clé gérée par le client AWS Key Management Service (AWS KMS) et d'accorder à Amazon Managed Service for Prometheus les autorisations requises pour accéder à la fois au secret et à sa clé de chiffrement. Les procédures suivantes vous guident à chaque étape de ce processus de configuration.

Pour créer un secret dans Secrets Manager pour PagerDuty

Pour l'utiliser PagerDuty comme récepteur d'alertes, vous devez enregistrer votre clé PagerDuty d'intégration dans Secrets Manager. Procédez comme suit :

1. Ouvrez la console Secrets Manager.
2. Choisissez Store a new secret (Stocker un nouveau secret).
3. Pour Secret type (Type de secret), choisissez Other type of secret (Autre type de secret).
4. Pour les paires clé/valeur, entrez votre clé PagerDuty d'intégration comme valeur secrète. Il s'agit de la clé de routage ou de la clé de service de votre PagerDuty intégration.
5. Choisissez Suivant.
6. Entrez le nom et la description de votre secret, puis choisissez Next.
7. Configurez les paramètres de rotation si vous le souhaitez, puis choisissez Next.
8. Vérifiez vos paramètres et choisissez Store.
9. Après avoir créé le secret, notez son ARN. Vous en aurez besoin pour configurer le gestionnaire d'alertes.

Pour chiffrer votre secret avec une clé gérée par le client AWS KMS

Vous devez autoriser Amazon Managed Service for Prometheus à accéder à votre secret et à sa clé de chiffrement :

1. Politique relative aux ressources secrètes : ouvrez votre secret dans la console Secrets Manager.
   a. Choisissez Autorisations relatives aux ressources.
   b. Choisissez Modifier les autorisations.
   c. Ajoutez la déclaration de politique suivante. Dans la déclaration, remplacez le *highlighted values* par vos valeurs spécifiques.

```
{
  "Effect": "Allow",
  "Principal": {
    "Service": "aps.amazonaws.com"
  },
  "Action": "secretsmanager:GetSecretValue",
  "Resource": "*",
  "Condition": {
    "ArnEquals": {
```

```
      "aws:SourceArn": "arn:aws:aps:aws-
region:123456789012:workspace/WORKSPACE_ID"
    },
    "StringEquals": {
      "aws:SourceAccount": "123456789012"
    }
  }
}
```

d. Choisissez Enregistrer.

2. Politique relative aux clés KMS : ouvrez votre AWS KMS clé dans la AWS KMS console.

   a. Choisissez Key policy.

   b. Choisissez Modifier.

   c. Ajoutez la déclaration de politique suivante. Dans la déclaration, remplacez le *highlighted values* par vos valeurs spécifiques.

```
{
  "Effect": "Allow",
  "Principal": {
    "Service": "aps.amazonaws.com"
  },
  "Action": "kms:Decrypt",
  "Resource": "*",
  "Condition": {
    "ArnEquals": {
      "aws:SourceArn": "arn:aws:aps:aws-
region:123456789012:workspace/WORKSPACE_ID"
    },
    "StringEquals": {
      "aws:SourceAccount": "123456789012"
    }
  }
}
```

   d. Choisissez Enregistrer.

Prochaines étapes — Passez à la rubrique suivante,Configurer le gestionnaire d'alertes pour envoyer des alertes à PagerDuty.

Configurer le gestionnaire d'alertes pour envoyer des alertes à PagerDuty

Pour configurer le gestionnaire d'alertes auquel envoyer des alertes PagerDuty, vous devez mettre à jour la définition de votre gestionnaire d'alertes. Pour ce faire, utilisez le Console de gestion AWS AWS CLI, ou AWS SDKs.

Example configuration du gestionnaire d'alertes

Voici un exemple de configuration du gestionnaire d'alertes qui envoie des alertes à PagerDuty. Dans l'exemple, remplacez le *highlighted values* par vos valeurs spécifiques.

```
alertmanager_config: |
  route:
    receiver: 'pagerduty-receiver'
    group_by: ['alertname']
    group_wait: 30s
    group_interval: 5m
    repeat_interval: 1h
  receivers:
    - name: 'pagerduty-receiver'
      pagerduty_configs:
      - routing_key:
          aws_secrets_manager:
            secret_arn: 'arn:aws:secretsmanager:aws-
region:123456789012:secret:YOUR_SECRET_NAME'
            secret_key: 'YOUR_SECRET_KEY'
            refresh_interval: 5m
        description: '{{ .CommonLabels.alertname }}'
        severity: 'critical'
        details:
          firing: '{{ .Alerts.Firing | len }}'
          status: '{{ .Status }}'
          instance: '{{ .CommonLabels.instance }}'
```

Example AWS CLI

Vous trouverez ci-dessous une AWS CLI commande utilisée pour mettre à jour la définition de votre gestionnaire d'alertes. Dans l'exemple, remplacez le *highlighted values* par vos valeurs spécifiques.

```
aws amp put-alert-manager-definition \
  --workspace-id WORKSPACE_ID \
  --data file://alertmanager-config.yaml
```

Résolution des problèmes PagerDuty d'intégration

Si aucune alerte n'est envoyée à PagerDuty, vérifiez les points suivants :

- Vérifiez que votre secret existe et qu'il contient la bonne clé PagerDuty d'intégration.
- Vérifiez que votre secret est chiffré à l'aide d'une clé KMS gérée par le client.
- Assurez-vous que les politiques de ressources relatives au secret et à la clé KMS accordent les autorisations nécessaires à Amazon Managed Service for Prometheus.
- Vérifiez que l'ARN de la configuration de votre gestionnaire d'alertes fait correctement référence à votre secret.
- Vérifiez que votre clé PagerDuty d'intégration est valide et active dans votre PagerDuty compte.

Amazon Managed Service for Prometheus prend en charge CloudWatch Amazon Logs et les indicateurs suivants pour faciliter CloudWatch le dépannage. Pour plus d'informations, consultez Surveillez les événements Prometheus via Amazon Managed Service grâce aux journaux CloudWatch et Utilisez CloudWatch des métriques pour surveiller les ressources Amazon Managed Service for Prometheus.

- `SecretFetchFailure`
- `AlertManagerNotificationsThrottledByIntegration`
- `AlertManagerNotificationsFailedByIntegration`

Téléchargez le fichier de configuration de votre gestionnaire d'alertes sur Amazon Managed Service for Prometheus

Une fois que vous savez ce que vous voulez inclure dans votre fichier de configuration d'Alert Manager, vous pouvez le créer et le modifier dans la console, ou vous pouvez télécharger un fichier existant avec la console Amazon Managed Service for Prometheus ou. AWS CLI

> **Note**
>
> Si vous utilisez un cluster Amazon EKS, vous pouvez également télécharger un fichier de configuration du gestionnaire d'alertes à l'aide de AWS Controllers for Kubernetes.

Pour utiliser la console Amazon Managed Service for Prometheus afin de modifier ou de remplacer la configuration de votre gestionnaire d'alertes

1. Ouvrez la console Amazon Managed Service for Prometheus à l'adresse. https://console.aws.amazon.com/prometheus/
2. Dans le coin supérieur gauche de la page, cliquez sur l'icône du menu, puis sur Tous les espaces de travail.
3. Choisissez l'ID de l'espace de travail, puis cliquez sur l'onglet Gestionnaire d'alertes.
4. Si l'espace de travail ne possède pas encore de définition de gestionnaire d'alertes, choisissez Add definition.

   > **Note**
   >
   > Si l'espace de travail possède une définition de gestionnaire d'alertes que vous souhaitez remplacer, choisissez plutôt Modifier.

5. Choisissez Choose file, sélectionnez le fichier de définition du gestionnaire d'alertes, puis cliquez sur Continuer.

   > **Note**
   >
   > Vous pouvez également créer un nouveau fichier et le modifier directement dans la console en choisissant l'option Créer une définition. Cela créera un exemple de configuration par défaut que vous modifierez avant le téléchargement.

AWS CLI Pour télécharger une configuration de gestionnaire d'alertes dans un espace de travail pour la première fois

1. Encodez en Base64 le contenu du fichier de votre gestionnaire d'alertes. Sous Linux, vous pouvez utiliser la commande suivante :

```
base64 input-file output-file
```

Sous macOS, vous pouvez utiliser la commande suivante :

```
openssl base64 input-file output-file
```

2. Pour télécharger le fichier, entrez l'une des commandes suivantes.

   Dans la AWS CLI version 2, entrez :

```
aws amp create-alert-manager-definition --data file://path_to_base_64_output_file
 --workspace-id my-workspace-id --region region
```

   Dans la AWS CLI version 1, entrez :

```
aws amp create-alert-manager-definition --data fileb://path_to_base_64_output_file
 --workspace-id my-workspace-id --region region
```

3. Il faut compter quelques secondes pour que la configuration de votre gestionnaire d'alertes soit activée. Pour vérifier l'état, entrez la commande suivante :

```
aws amp describe-alert-manager-definition --workspace-id workspace_id --
region region
```

   Si l'`status` est `ACTIVE`, la nouvelle définition de votre gestionnaire d'alertes a pris effet.

Pour utiliser le AWS CLI pour remplacer la configuration du gestionnaire d'alertes d'un espace de travail par une nouvelle

1. Encodez en Base64 le contenu du fichier de votre gestionnaire d'alertes. Sous Linux, vous pouvez utiliser la commande suivante :

```
base64 input-file output-file
```

   Sous macOS, vous pouvez utiliser la commande suivante :

```
openssl base64 input-file output-file
```

2. Pour télécharger le fichier, entrez l'une des commandes suivantes.

   Dans la AWS CLI version 2, entrez :

   ```
   aws amp put-alert-manager-definition --data file://path_to_base_64_output_file --
   workspace-id my-workspace-id --region region
   ```

   Dans la AWS CLI version 1, entrez :

   ```
   aws amp put-alert-manager-definition --data file://path_to_base_64_output_file --
   workspace-id my-workspace-id --region region
   ```

3. Il faut compter quelques secondes pour que votre nouvelle configuration de gestionnaire d'alertes soit activée. Pour vérifier l'état, entrez la commande suivante :

   ```
   aws amp describe-alert-manager-definition --workspace-id workspace_id --
   region region
   ```

   Si l'`status` est `ACTIVE`, la nouvelle définition de votre gestionnaire d'alertes a pris effet. Jusqu'à ce moment, la configuration précédente de votre gestionnaire d'alertes reste active.

Intégrez des alertes à Amazon Managed Grafana ou à Grafana open source

Les règles d'alerte que vous avez créées dans le gestionnaire d'alertes au sein d'Amazon Managed Service for Prometheus peuvent être transmises et consultées dans Amazon Managed Grafana et Grafana, unifiant ainsi vos règles d'alerte et vos alertes dans un environnement unique. Dans Amazon Managed Grafana, vous pouvez consulter vos règles d'alerte et les alertes générées.

Conditions préalables

Avant de commencer à intégrer Amazon Managed Service for Prometheus dans Amazon Managed Grafana, vous devez remplir les conditions suivantes :

- Vous devez disposer d'informations d'identification Compte AWS et d'identification IAM existantes pour créer des rôles Amazon Managed Service for Prometheus et IAM par programmation.

Pour plus d'informations sur la création d'informations d'identification Compte AWS et IAM, consultezConfiguration AWS.

- Vous devez disposer d'un espace de travail Amazon Managed Service for Prometheus et y ingérer des données. Pour configurer un nouvel espace de travail, consultez la section Création d'un espace de travail Amazon Managed Service for Prometheus. Vous devez également connaître les concepts de Prometheus tels que le gestionnaire d'alertes et les règles. Pour plus d'informations sur ces rubriques, consultez la documentation Prometheus.
- Vous disposez d'une configuration de gestionnaire d'alertes et d'un fichier de règles déjà configurés dans Amazon Managed Service for Prometheus. Pour plus d'informations sur le gestionnaire d'alertes dans Amazon Managed Service for Prometheus, consultez la section Gestion et transfert des alertes dans Amazon Managed Service for Prometheus avec le gestionnaire d'alertes. Pour plus d'informations sur les règles, consultez Utilisation de règles pour modifier ou surveiller les métriques au fur et à mesure de leur réception.
- Amazon Managed Grafana doit être installé ou la version open source de Grafana doit être en cours d'exécution.
  - Si vous utilisez Amazon Managed Grafana, vous devez utiliser les alertes Grafana. Pour plus d'informations, consultez la section Migrating legacy dashboard alerts to Grafana alerting.
  - Si vous utilisez la version open source de Grafana, vous devez utiliser la version 9.1 ou une version supérieure.

    > **Note**
    >
    > Vous pouvez utiliser les versions antérieures de Grafana, mais vous devez activer la fonctionnalité d'alerte unifiée (alerte Grafana), et vous devrez peut-être configurer un proxy sigv4 pour passer des appels depuis Grafana vers Amazon Managed Service for Prometheus. Pour de plus amples informations, veuillez consulter Configuration de Grafana open source ou Grafana Enterprise pour une utilisation avec Amazon Managed Service for Prometheus.

- Amazon Managed Grafana doit avoir les autorisations suivantes pour vos ressources Prometheus. Vous devez les ajouter aux politiques gérées par le service ou aux politiques gérées par le client décrites dans https://docs.aws.amazon.com/grafana/latest/userguide/AMG-manage-permissions.html.
  - `aps:ListRules`
  - `aps:ListAlertManagerSilences`

- `aps:ListAlertManagerAlerts`
- `aps:GetAlertManagerStatus`
- `aps:ListAlertManagerAlertGroups`
- `aps:PutAlertManagerSilences`
- `aps:DeleteAlertManagerSilence`

Configuration d'Amazon Managed Grafana

Si vous avez déjà défini des règles et des alertes dans votre instance Amazon Managed Service for Prometheus, la configuration permettant d'utiliser Amazon Managed Grafana comme tableau de bord pour ces alertes est entièrement effectuée dans Amazon Managed Grafana.

Pour configurer Amazon Managed Grafana comme tableau de bord des alertes

1. Ouvrez la console Grafana pour votre espace de travail.
2. Sous Configurations, sélectionnez Sources de données.
3. Créez ou ouvrez votre source de données Prometheus. Si vous n'avez pas encore configuré de source de données Prometheus, consultez la section Étape 2 : Ajouter la source de données Prometheus dans Grafana pour plus d'informations.
4. Dans la source de données Prometheus, sélectionnez Manage alerts via Alertmanager UI.
5. Revenez à l'interface Sources de données.
6. Créez une nouvelle source de données de gestionnaire d'alertes.
7. Sur la page de configuration de la source de données du gestionnaire d'alertes, ajoutez les paramètres suivants :

   - Définissez Implementation sur `Prometheus`.
   - Pour le paramètre URL, utilisez l'URL de votre espace de travail Prometheus, supprimez tout ce qui se trouve après l'ID d'espace de travail et ajoutez `/alertmanager` à la fin. Dans l'exemple suivant, remplacez les *`variables`* par vos propres informations (spécifiques au compte) :

     ```
     https://aps-workspaces.US East (N. Virginia).amazonaws.com/workspaces/ws-
     example-1234-5678-abcd-xyz00000001/alertmanager.
     ```

- Sous Auth, activez SigV4Auth. Cela indique à Grafana d'utiliser l'authentificationAWS pour les demandes.
- Sous Sigv4Auth Details, pour Default Region, indiquez la région de votre instance Prometheus, par exemple `us-east-1`.
- Définissez l'option Default sur `true`.

8. Choisissez Save and test.
9. Vos alertes Amazon Managed Service for Prometheus doivent désormais être configurées pour fonctionner avec votre instance Grafana. Sur la page Alerte de Grafana, vérifiez que vous pouvez voir Alert rules, Alert groups (notamment les alertes actives) et Silences sur votre instance Amazon Managed Service for Prometheus.

Résoudre les problèmes liés au gestionnaire d'alertes avec les journaux CloudWatch

Surveillez les événements Prometheus via Amazon Managed Service grâce aux journaux CloudWatch vous permet de résoudre les problèmes liés au gestionnaire d'alertes et à l'outil de règle. Cette section contient des rubriques de dépannage relatives au gestionnaire d'alertes.

Rubriques

- Alertes actives (avertissement)
- Avertissement concernant la taille du groupe d'agrégation d'alertes
- La taille des alertes est trop grande (avertissement)
- Avertissement de contenu vide
- Avertissement key/value non valide
- Avertissement de limite de message
- Aucune erreur de stratégie basée sur les ressources
- Avertissement de format non ASCII
- Non autorisé à appeler KMS
- Erreur de modèle

Alertes actives (avertissement)

Lorsque le journal contient l'avertissement suivant

```
{
    "workspaceId": "ws-efdc5b42-b051-11ec-b123-4567ac120002",
    "message": {
        "log": "too many alerts, limit: 1000",
        "level": "WARN"
    },
    "component": "alertmanager"
}
```

Cela signifie que le quota d'alertes actives du gestionnaire d'alertes est dépassé.

Action à exécuter

Demandez une augmentation de quota. Connectez-vous à la console Service Quotas Console de gestion AWS et ouvrez-la à l'adresse https://console.aws.amazon.com/servicequotas/.

Avertissement concernant la taille du groupe d'agrégation d'alertes

Lorsque le journal contient l'avertissement suivant

```
{
    "workspaceId": "ws-efdc5b42-b051-11ec-b123-4567ac120002",
    "message": {
        "log": "Too many aggregation groups, cannot create new group for alert,
 groups=1000, limit=1000, alert=sample-alert",
        "level": "WARN"
    },
    "component": "alertmanager"
}
```

Cela signifie que le quota de taille du groupe d'agrégation d'alertes du gestionnaire d'alertes a été dépassé.

Action à exécuter

Réduisez la taille du groupe d'agrégation d'alertes en utilisant le `group_by` paramètre. Pour plus d'informations, consultez la section Paramètres relatifs à l'itinéraire dans la documentation de Prometheus.

Vous pouvez également demander une augmentation de quota. Connectez-vous à la console Service Quotas Console de gestion AWS et ouvrez-la à l'adresse https://console.aws.amazon.com/servicequotas/.

La taille des alertes est trop grande (avertissement)

Lorsque le journal contient l'avertissement suivant

```
{
    "workspaceId": "ws-efdc5b42-b051-11ec-b123-4567ac120002",
    "message": {
        "log": "alerts too big, total size limit: 20000000 bytes",
        "level": "WARN"
    },
    "component": "alertmanager"
}
```

Cela signifie que le quota d'alertes par espace de travail, en termes de taille, a été dépassé.

Action à exécuter

Supprimez les annotations et les étiquettes inutiles afin de réduire la taille des alertes.

Avertissement de contenu vide

Lorsque le journal contient l'avertissement suivant

```
{
   "workspaceId": "ws-abcd1234-ef56-78ab-cd90-1234abcd0000",
   "message": {
       "log": "Message has been modified because the content was empty."
       "level": "WARN"
   },
   "component": "alertmanager"
}
```

Cela signifie que le modèle de gestionnaire d'alertes a résolu l'alerte sortante en message vide.

Action à exécuter

Validez votre modèle de gestionnaire d'alertes et assurez-vous que vous disposez d'un modèle valide pour tous les chemins de réception.

Avertissement **`key/value`** non valide

Lorsque le journal contient l'avertissement suivant

```
{
   "workspaceId": "ws-abcd1234-ef56-78ab-cd90-1234abcd0000",
   "message": {
       "log": "MessageAttributes has been removed because of invalid key/value,
 numberOfRemovedAttributes=1"
       "level": "WARN"
   },
   "component": "alertmanager"
}
```

Cela signifie que certains attributs du message ont été supprimés car ils n' keys/values étaient pas valides.

Action à exécuter

Réévaluez les modèles que vous utilisez pour renseigner les attributs des messages et assurez-vous qu'ils correspondent à des attributs de message SNS valides. Pour plus d'informations sur la validation d'un message envoyé à une rubrique Amazon SNS, consultez la section Validating SNS topic.

Avertissement de limite de message

Lorsque le journal contient l'avertissement suivant

```
{
   "workspaceId": "ws-abcd1234-ef56-78ab-cd90-1234abcd0000",
   "message": {
       "log": "Message has been truncated because it exceeds size limit,
 originSize=266K, truncatedSize=12K"
       "level": "WARN"
   },
   "component": "alertmanager"
}
```

Cela signifie qu'une partie de la taille du message est trop grande.

Action à exécuter

Examinez le modèle de message du récepteur d'alertes et modifiez-le pour qu'il respecte la limite de taille.

Aucune erreur de stratégie basée sur les ressources

Lorsque le journal contient l'erreur suivante

```
{
   "workspaceId": "ws-abcd1234-ef56-78ab-cd90-1234abcd0000",
   "message": {
       "log": "Notify for alerts failed, AMP is not authorized to perform: SNS:Publish
 on resource: arn:aws:sns:us-west-2:12345:testSnsReceiver because no resource-based
 policy allows the SNS:Publish action"
       "level": "ERROR"
   },
   "component": "alertmanager"
}
```

Cela signifie qu'Amazon Managed Service for Prometheus n'est pas autorisé à envoyer l'alerte à la rubrique SNS spécifiée.

Action à exécuter

Vérifiez que la stratégie d'accès de la rubrique SNS autorise Amazon Managed Service for Prometheus à envoyer des messages SNS à la rubrique. Créez une politique d'accès aux réseaux sociaux donnant au service `aps.amazonaws.com` (Amazon Managed Service for Prometheus) l'accès à votre rubrique Amazon SNS. Pour plus d'informations sur les politiques d'accès aux réseaux sociaux, consultez les sections Utilisation du langage des politiques d'accès et exemples de cas pour le contrôle d'accès Amazon SNS dans le guide du développeur Amazon Simple Notification Service.

Avertissement de format non ASCII

Lorsque le journal contient l'avertissement suivant

```
{
   "workspaceId": "ws-abcd1234-ef56-78ab-cd90-1234abcd0000",
   "message": {
       "log": "Subject has been modified because it contains control or non-ASCII
 characters."
       "level": "WARN"
   },
   "component": "alertmanager"
}
```

Cela signifie que l'objet comporte des caractères non ASCII.

Action à exécuter

Dans le champ d'objet de votre modèle, supprimez les références aux étiquettes susceptibles de contenir des caractères non ASCII.

Non autorisé à appeler KMS

Lorsque le journal contient l' AWS KMS erreur suivante

```
{
   "workspaceId": "ws-abcd1234-ef56-78ab-cd90-1234abcd0000",
   "message": {
       "log": "Notify for alerts failed, AMP is not authorized to call KMS",
       "level": "ERROR"
   },
   "component": "alertmanager"
}
```

Action à exécuter

Vérifiez que la politique clé de la clé utilisée pour chiffrer le sujet Amazon SNS autorise le `aps.amazonaws.com` principal de service Amazon Managed Service for Prometheus à effectuer les actions suivantes : et. `kms:GenerateDataKey*` `kms:Decrypt` Pour plus d'informations, veuillez consulter la section AWS Autorisations KMS pour SNS.

Erreur de modèle

Lorsque le journal contient l'erreur suivante

```
             {
  "workspaceId": "ws-efdc5b42-b051-11ec-b123-4567ac120002",
  "message": {
      "log": "Notify for alerts failed. There is an error in a receiver that is using
templates in the AlertManager definition. Make sure that the syntax is correct and
only template functions and variables that exist are used in the receiver 'default',
sns_configs position #2, section 'attributes'"
      "level": "ERROR"
  },
  "component": "alertmanager"
```

```
}
```

Cela signifie qu'un modèle utilisé dans la AlertManager définition contient une erreur. L'entrée d'erreur contient des instructions concernant le récepteur, la position dans le fichier sns_configs et la propriété contenant les erreurs.

Action à exécuter

Validez votre définition d'Alert Manager. Assurez-vous que la syntaxe est correcte et que vous faites référence aux variables de modèle et aux fonctions existantes. Pour plus d'informations, consultez la référence du modèle de notification dans la documentation open source de Prometheus.

Journalisation et surveillance d'Amazon Managed Service pour les espaces de travail Prometheus

Amazon Managed Service for Prometheus utilise CloudWatch Amazon pour fournir des données sur son fonctionnement. Vous pouvez utiliser CloudWatch les statistiques pour en savoir plus sur l'utilisation des ressources et les demandes adressées à votre Amazon Managed Service pour les espaces de travail Prometheus. Vous pouvez activer la prise en charge CloudWatch des journaux pour obtenir des journaux des événements qui se produisent dans vos espaces de travail.

Les rubriques suivantes décrivent l'utilisation CloudWatch de manière plus détaillée.

Utilisez CloudWatch des métriques pour surveiller les ressources Amazon Managed Service for Prometheus

Amazon Managed Service for Prometheus envoie des statistiques d'utilisation à. CloudWatch Ces métriques fournissent une visibilité sur l'utilisation de votre espace de travail. Les métriques vendues se trouvent dans les `AWS/Prometheus` espaces de noms `AWS/Usage` et dans. CloudWatch Ces statistiques sont disponibles CloudWatch gratuitement. Pour plus d'informations sur les métriques, consultez Métriques d'utilisation d'CloudWatch .

CloudWatch nom de la métrique	Nom de la ressource	CloudWatch espace de noms	Description
ResourceCount*	CreateAlertManagerAlertsTPS	`AWS/Usage`	Le nombre maximum d'opérations d'`CreateAlertManagerAlerts` API par seconde, par espace de travail
ResourceCount*	DeleteAlertManagerSilencesTPS	`AWS/Usage`	Le nombre maximum d'opérations d'`DeleteAlertManagerSilences` API par seconde, par espace de travail

CloudWatch nom de la métrique	Nom de la ressource	CloudWatch espace de noms	Description
ResourceCount*	GetAlertManagerSilenceTPS	AWS/Usage	Le nombre maximum d'opérations d'GetAlertManagerSilence API par seconde, par espace de travail
ResourceCount*	GetAlertManagerStatusTPS	AWS/Usage	Le nombre maximum d'opérations d'GetAlertManagerStatus API par seconde, par espace de travail
ResourceCount*	GetLabelsTPS	AWS/Usage	Le nombre maximum d'opérations d'GetLabels API par seconde, par espace de travail
ResourceCount*	GetMetricMetadataTPS	AWS/Usage	Le nombre maximum d'opérations d'GetMetricMetadata API par seconde, par espace de travail
ResourceCount*	GetSeriesTPS	AWS/Usage	Le nombre maximum d'opérations d'GetSeries API par seconde, par espace de travail
ResourceCount	InhibitionRulesInAlertManagerDefinition	AWS/Usage	Nombre maximal de règles d'inhibition dans le fichier de définition du gestionnaire d'alertes.

CloudWatch nom de la métrique	Nom de la ressource	CloudWatch espace de noms	Description
ResourceCount*	ListAlertManagerAlertGroupInfosTPS	AWS/Usage	Le nombre maximum d'opérations d'`ListAlertManagerAlertGroupInfos` API par seconde, par espace de travail
ResourceCount*	ListAlertManagerAlertGroupsTPS	AWS/Usage	Le nombre maximum d'opérations d'`ListAlertManagerAlertGroups` API par seconde, par espace de travail
ResourceCount*	ListAlertManagerAlertsTPS	AWS/Usage	Le nombre maximum d'opérations d'`ListAlertManagerAlerts` API par seconde, par espace de travail
ResourceCount*	ListAlertManagerReceiversTPS	AWS/Usage	Le nombre maximum d'opérations d'`ListAlertManagerReceivers` API par seconde, par espace de travail
ResourceCount*	ListAlertManagerSilencesTPS	AWS/Usage	Le nombre maximum d'opérations d'`ListAlertManagerSilences` API par seconde, par espace de travail

CloudWatch nom de la métrique	Nom de la ressource	CloudWatch espace de noms	Description
ResourceCount*	ListAlertsTPS	AWS/Usage	Le nombre maximum d'opérations d'ListAlerts API par seconde, par espace de travail
ResourceCount*	ListRulesTPS	AWS/Usage	Le nombre maximum d'opérations d'ListRules API par seconde, par espace de travail
ResourceCount*	PutAlertManagerSilencesTPS	AWS/Usage	Le nombre maximum d'opérations d'PutAlertManagerSilences API par seconde, par espace de travail
ResourceCount	HAReplicaGroupCount	AWS/Usage	Nombre de groupes de répliques à haute disponibilité
ResourceCount*	QueryMetricsTPS	AWS/Usage	Opérations de requête par seconde
ResourceCount*	RemoteWriteTPS	AWS/Usage	Opérations d'écriture à distance par seconde
ResourceCount	ActiveAlerts	AWS/Usage	Nombre d'alertes actives par espace de travail Unités : nombre Statistiques valides : moyenne, minimale, maximale

CloudWatch nom de la métrique	Nom de la ressource	CloudWatch espace de noms	Description
ResourceCount	ActiveSeries	AWS/Usage	Nombre de séries actives par espace de travail Unités : nombre Statistiques valides : moyenne, minimale, maximale
ResourceCount	AlertAggregationGroupSize	AWS/Usage	Taille maximale d'un groupe d'agrégation d'alertes dans le fichier de définition du gestionnaire d'alertes. Chaque combinaison de valeurs d'étiquette group_by créerait un groupe d'agrégation.
ResourceCount	AlertManagerDefinitionSizeBytes	AWS/Usage	Taille maximale d'un fichier de définition du gestionnaire d'alertes, en octets.
ResourceCount	AllSilences	AWS/Usage	Nombre maximum de silences, y compris les silences expirés, actifs et en attente, par espace de travail.

CloudWatch nom de la métrique	Nom de la ressource	CloudWatch espace de noms	Description
ResourceCount	IngestionRate	AWS/Usage	Taux d'ingestion d'échantillons Unités : nombre par seconde Statistiques valides : moyenne, minimale, maximale
ResourceCount	RuleEvaluationInterval	AWS/Usage	Intervalle minimal d'évaluation des règles
ResourceCount	RuleGroupNamespaceDefinitionSizeBytes	AWS/Usage	Taille maximale d'un fichier de définition d'espace de noms de groupe de règles, en octets.
ResourceCount	TemplatesInAlertManagerDefinition	AWS/Usage	Nombre maximal de modèles dans le fichier de définition du gestionnaire d'alertes.
ResourceCount	WorkspaceCount	AWS/Usage	Le nombre maximum d'espaces de travail par région, par compte.
ResourceCount	SizeOfAlerts	AWS/Usage	Taille totale de toutes les alertes de l'espace de travail, en octets Unités : octets Statistiques valides : moyenne, minimale, maximale

CloudWatch nom de la métrique	Nom de la ressource	CloudWatch espace de noms	Description
ResourceCount	SuppressedAlerts	AWS/Usage	Nombre d'alertes supprimées par espace de travail. Une alerte peut être supprimée par un silence ou une inhibition. Unités : nombre Statistiques valides : moyenne, minimale, maximale
ResourceCount	UnprocessedAlerts	AWS/Usage	Nombre d'alertes non traitées par espace de travail. Une alerte n'est pas traitée une fois reçue par AlertManager, mais elle attend la prochaine évaluation du groupe d'agrégation. Unités : nombre Statistiques valides : moyenne, minimale, maximale
ResourceCount	AllAlerts	AWS/Usage	Nombre d'alertes par espace de travail, quel que soit l'état Unités : nombre Statistiques valides : moyenne, minimale, maximale

CloudWatch nom de la métrique	Nom de la ressource	CloudWatch espace de noms	Description
ResourceCount	AllRules	AWS/Usage	Nombre de règles par état par espace de travail Unités : nombre Statistiques valides : moyenne, minimale, maximale
ResourceCount	NativeHistogramActiveSeries	AWS/Usage	Le nombre de séries actives d'histogrammes natifs uniques par espace de travail Unités : nombre Statistiques valides : moyenne, minimale, maximale
ResourceCount	NativeHistogramIngestionRate	AWS/Usage	Taux d'ingestion d'échantillons d'histogrammes natifs par espace de travail et par seconde Unités : nombre par seconde Statistiques valides : moyenne, minimale, maximale

CloudWatch nom de la métrique	Nom de la ressource	CloudWatch espace de noms	Description
ActiveSeriesPerLabelSet	-	`AWS/Prometheus`	L'utilisation actuelle de la série active pour chaque jeu d'étiquettes défini par l'utilisateur Unités : nombre Statistiques valides : moyenne, minimum, maximum, somme
ActiveSeriesLimitPerLabelSet	-	`AWS/Prometheus`	La valeur limite de série active actuelle pour chaque jeu d'étiquettes défini par l'utilisateur Unités : nombre Statistiques valides : moyenne, minimum, maximum, somme
AlertManagerAlerts Received	-	`AWS/Prometheus`	Nombre total d'alertes réussies reçues par le gestionnaire d'alertes Unités : nombre Statistiques valides : moyenne, minimum, maximum, somme

CloudWatch nom de la métrique	Nom de la ressource	CloudWatch espace de noms	Description
AlertManagerNotificationsFailed	-	AWS/Prometheus	Nombre de livraisons d'alertes ayant échoué Unités : nombre Statistiques valides : moyenne, minimum, maximum, somme
AlertManagerNotificationsThrottled	-	AWS/Prometheus	Nombre d'alertes bloquées Unités : nombre Statistiques valides : moyenne, minimum, maximum, somme
AnomalyDetectors	WorkspaceId	AWS/Prometheus	Nombre total de détecteurs d'anomalies pour un espace de travail donné Unités : nombre Statistiques valides : moyenne, minimale, maximale
AnomalyDetectorEvaluations	WorkspaceId, AnomalyDetectorId	AWS/Prometheus	Nombre total d'évaluations de détecteurs d'anomalies Unités : nombre Statistiques valides : moyenne, minimum, maximum, somme

CloudWatch nom de la métrique	Nom de la ressource	CloudWatch espace de noms	Description
AnomalyDetectorEvaluationFailures	WorkspaceId, AnomalyDetectorId	AWS/Prometheus	Nombre de défaillances des détecteurs d'anomalies dans l'intervalle Unités : nombre Statistiques valides : moyenne, minimum, maximum, somme
AnomalyDetectorLastEvaluationDuration	WorkspaceId, AnomalyDetectorId	AWS/Prometheus	Durée de la dernière évaluation d'un détecteur d'anomalies Unités : secondes Statistiques valides : moyenne, minimum, maximum, somme
AnomalyDetectorMissedEvaluations	WorkspaceId, AnomalyDetectorId	AWS/Prometheus	Nombre d'évaluations de détecteurs d'anomalies manquées dans l'intervalle Unités : nombre Statistiques valides : moyenne, minimum, maximum, somme

CloudWatch nom de la métrique	Nom de la ressource	CloudWatch espace de noms	Description
Discarded Samples**	-	AWS/Prometheus	Nombre d'échantillons rejetés par motif Unités : nombre Statistiques valides : moyenne, minimum, maximum, somme
Discarded Series**	-	AWS/Prometheus	Nombre de séries contenant un échantillon jeté pour une raison Unités : nombre Statistiques valides : moyenne, minimum, maximum, somme
DiscardedSamplesPerLabelSet	-	AWS/Prometheus	Le nombre d'échantillons rejetés pour chaque jeu d'étiquettes défini par l'utilisateur Unités : nombre Statistiques valides : moyenne, minimum, maximum, somme

CloudWatch nom de la métrique	Nom de la ressource	CloudWatch espace de noms	Description
DiscardedSeriesPerLabelSet	-	AWS/Prometheus	Nombre de séries contenant un échantillon jeté pour chaque jeu d'étiquettes défini par l'utilisateur Unités : nombre Statistiques valides : moyenne, minimum, maximum, somme
IngestionRatePerLabelSet	-	AWS/Prometheus	Le taux d'ingestion pour chaque jeu d'étiquettes défini par l'utilisateur Unités : nombre Statistiques valides : moyenne, minimum, maximum, somme
NativeHistogramIngestedBucketsRate	-	AWS/Prometheus	Taux de compartiments remplis reçus par seconde sur tous les échantillons d'histogrammes natifs. Exclut les seaux rejetés. Unités : nombre par seconde Statistiques valides : moyenne, minimale, maximale

CloudWatch nom de la métrique	Nom de la ressource	CloudWatch espace de noms	Description
NativeHistogramReducedResolutionCount	-	`AWS/Prometheus`	Nombre d'échantillons d'histogrammes natifs avec une résolution de compartiment automatiquement réduite. La résolution est réduite lorsqu'un échantillon dépasse la limite maximale du nombre de compartiments. Unités : nombre Statistiques valides : moyenne, minimum, maximum, somme
OutOfOrderIngestionRate	-	`AWS/Prometheus`	Out-of-order taux d'ingestion d'échantillons Unités : nombre par seconde Statistiques valides : moyenne, minimum, maximum, somme

CloudWatch nom de la métrique	Nom de la ressource	CloudWatch espace de noms	Description
OutOfOrderSampleAge***	-	AWS/Prometheus	Différence entre l'horodatage d'un échantillon en rupture de commande et son heure d'ingestion, qui indique l'âge de l'échantillon au moment de son ingestion. Unités : secondes Statistiques valides : moyenne, minimum, maximum, somme
QuerySamplesProcessed	-	AWS/Prometheus	Nombre d'échantillons de requêtes traités Unités : nombre Statistiques valides : moyenne, minimum, maximum, somme
RuleEvaluations	-	AWS/Prometheus	Nombre total d'évaluations de règles Unités : nombre Statistiques valides : moyenne, minimum, maximum, somme

CloudWatch nom de la métrique	Nom de la ressource	CloudWatch espace de noms	Description
RuleEvaluationFailures	-	AWS/Prometheus	Nombre d'échecs d'évaluation des règles dans l'intervalle Unités : nombre Statistiques valides : moyenne, minimum, maximum, somme
RuleGroupIterationsMissed	-	AWS/Prometheus	Nombre d'itérations de groupes de règles manquées dans l'intervalle. Unités : nombre Statistiques valides : moyenne, minimum, maximum, somme
RuleGroupLastEvaluationDuration	-	AWS/Prometheus	Durée de la dernière évaluation d'un groupe de règles. Unités : secondes Statistiques valides : moyenne, minimum, maximum, somme

* Les métriques TPS sont générées chaque minute et sont une moyenne par seconde sur cette minute. Les courtes périodes de rafale ne seront pas capturées dans les métriques TPS.

** Certaines des raisons pour lesquelles les échantillons sont jetés sont les suivantes. Les raisons ci-dessous n'apparaissent pas toutes dans la DiscardedSeries métrique.

Raison	Signification
greater_than_max_sample_age	Jeter les échantillons datant de plus d'une heure.
new-value-for-timestamp	Les échantillons dupliqués sont envoyés avec le même horodatage que l'échantillon précédent, mais avec des valeurs différentes.
per_labelset_series_limit	L'utilisateur a atteint le nombre total de séries actives par limite définie d'étiquettes.
per_metric_series_limit	L'utilisateur a atteint la limite de séries actives par métrique.
per_user_series_limit	L'utilisateur a atteint la limite du nombre total de séries actives.
rate_limited	Taux d'ingestion limité.
sample-out-of-order	Les échantillons sont envoyés hors commande et ne peuvent pas être traités.
échantillon trop vieux	L'échantillon est plus ancien que la période de rupture configurée et ne peut pas être traité.
limite de taux de commande hors limite	Out-of-order la limite de taux d'ingestion a été atteinte et l'échantillon ne peut pas être traité. Pour de plus amples informations, veuillez consulter Service Quotas d'Amazon Managed Service for Prometheus.
label_value_too long	La valeur de l'étiquette est supérieure à la limite de caractères autorisée.
max_label_names_per_series	L'utilisateur a sélectionné les noms des libellés par métrique.
missing_metric_name	Le nom de la métrique n'est pas fourni.
metric_name_invalid	Nom de métrique non valide fourni.

Raison	Signification
label_invalid	Libellé fourni non valide.
duplicate_label_names	Des noms d'étiquettes dupliqués ont été fournis.
histogramme_native_sample_size_bytes_exceeded	L'échantillon d'histogramme natif dépasse la taille d'échantillon maximale autorisée en octets.
Schéma d'histogramme natif non valide	La valeur de schéma de l'histogramme natif n'est pas valide. Les schémas valides sont compris entre -4 et 8.
histogramme_natif invalide	L'histogramme natif échoue à la validation (par exemple, nombre de compartiments négatif, nombre de compartiments non concordant ou travées de compartiments mal formées).
native_histogram_buckets_exceeded	L'histogramme natif dépasse le nombre maximal de compartiments et la résolution ne peut pas être réduite automatiquement.
native_histogram_rate_limited	L'échantillon d'histogramme natif a été rejeté car la limite de taux d'ingestion de l'histogramme natif a été atteinte.
par utilisateur_native_histogram_series_limit	L'utilisateur a atteint la limite de séries actives de l'histogramme natif par espace de travail.

*** La `OutOfOrderSampleAge` métrique inclut une `Percentile` dimension. Vous pouvez utiliser cette métrique pour déterminer le créneau horaire hors service approprié pour votre espace de travail. Les valeurs valides pour la `Percentile` dimension sont les suivantes.

Percentile	Description
p50	L'âge du 50e centile des échantillons en rupture de commande.
p99	L'âge du 99e centile des échantillons en rupture de commande.

Percentile	Description
max	Âge maximum des échantillons en rupture de commande.

> **Note**
>
> Une métrique inexistante ou manquante est identique à la valeur de cette métrique égale à 0.

> **Note**
>
> `RuleGroupIterationsMissed`, `RuleEvaluationsRuleEvaluationFailures`, et `RuleGroupLastEvaluationDuration` ont la `RuleGroup` dimension de la structure suivante :
> *`RuleGroupNamespace`*;*`RuleGroup`*

Régler une CloudWatch alarme sur les métriques vendues par Prometheus

Vous pouvez surveiller l'utilisation des ressources Prometheus à l'aide d'alarmes. CloudWatch

Pour régler une alarme sur le nombre de ActiveSeries dans Prometheus

1. Choisissez l'onglet Graphed metrics et faites défiler l'écran vers le bas jusqu'à l'ActiveSeriesétiquette.

   Dans la vue Graphed metrics, seules les métriques actuellement ingérées apparaissent.
2. Sélectionnez l'icône de notification dans la colonne Actions.
3. Dans Specify metric and conditions, entrez la condition de seuil dans le champ Conditions value et choisissez Suivant.
4. Dans Configure actions, sélectionnez une rubrique SNS existante ou créez-en une nouvelle à laquelle envoyer la notification.
5. Dans Add name and description, ajoutez le nom de l'alarme et une description facultative.
6. Sélectionnez Créer une alerte.

Surveillez les événements Prometheus via Amazon Managed Service grâce aux journaux CloudWatch

Amazon Managed Service for Prometheus enregistre les événements d'erreur et d'avertissement d'Alert Manager et de Ruler dans des groupes de journaux dans Amazon Logs. CloudWatch Pour plus d'informations sur le gestionnaire d'alertes et les règles, consultez la section Gestionnaire d'alertes du présent guide. Vous pouvez publier les données des journaux de l'espace de travail dans les flux de CloudWatch journaux dans Logs. Vous pouvez configurer les journaux que vous souhaitez surveiller dans la console Amazon Managed Service for Prometheus ou en utilisant l' AWS CLI. Vous pouvez consulter ou interroger ces journaux dans la CloudWatch console. Pour plus d'informations sur l'affichage CloudWatch des flux de journaux dans la console, consultez la section Utilisation des groupes de journaux et des flux de journaux CloudWatch dans le guide de CloudWatch l'utilisateur.

Le niveau CloudWatch gratuit permet de publier jusqu'à 5 Go de CloudWatch journaux dans Logs. Les journaux qui dépassent la limite du niveau gratuit seront facturés sur la base du plan CloudWatch tarifaire.

Rubriques

- Configuration des CloudWatch journaux

Configuration des CloudWatch journaux

Amazon Managed Service for Prometheus enregistre les événements d'erreur et d'avertissement d'Alert Manager et de Ruler dans des groupes de journaux dans Amazon Logs. CloudWatch

Vous pouvez définir la configuration de journalisation des CloudWatch journaux dans la console Amazon Managed Service for Prometheus ou en appelant AWS CLI `create-logging-configuration` la demande d'API.

Conditions préalables

Avant d'appeler`create-logging-configuration`, associez la politique suivante ou des autorisations équivalentes à l'ID ou au rôle que vous utiliserez pour configurer CloudWatch les journaux.

JSON

```
{
```

```
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogDelivery",
                "logs:GetLogDelivery",
                "logs:UpdateLogDelivery",
                "logs:DeleteLogDelivery",
                "logs:ListLogDeliveries",
                "logs:PutResourcePolicy",
                "logs:DescribeResourcePolicies",
                "logs:DescribeLogGroups",
                "aps:CreateLoggingConfiguration",
                "aps:UpdateLoggingConfiguration",
                "aps:DescribeLoggingConfiguration",
                "aps:DeleteLoggingConfiguration"
            ],
            "Resource": "*"
        }
    ]
}
```

Pour configurer les CloudWatch journaux

Vous pouvez configurer la connexion à Amazon Managed Service pour Prometheus à l'aide de la console ou AWS du. AWS CLI

Console

Pour configurer la journalisation dans la console Amazon Managed Service for Prometheus

1. Accédez à l'onglet Journaux dans le volet des détails de votre espace de travail.
2. Choisissez Manage logs dans le coin supérieur droit du volet Journaux.
3. Choisissez Tout dans la liste déroulante Niveau de journalisation.
4. Choisissez le groupe de journaux dans lequel vous souhaitez publier vos journaux dans la liste déroulante Log Group.

   Vous pouvez également créer un nouveau groupe de journaux dans CloudWatch la console.
5. Sélectionnez Enregistrer les modifications.

AWS CLI

Vous pouvez définir la configuration de journalisation à l'aide du AWS CLI.

Pour configurer la journalisation à l'aide du AWS CLI

- À l'aide de AWS CLI, exécutez la commande suivante.

```
aws amp create-logging-configuration --workspace-id my_workspace_ID
                                     --log-group-arn my-log-group-arn
```

Limitations

- Tous les événements ne sont pas consignés

  Amazon Managed Service for Prometheus consigne uniquement les événements de niveau `warning` ou `error`.

- Limites de taille de politique

  CloudWatch Les politiques relatives aux ressources des journaux sont limitées à 5 120 caractères. Lorsque les CloudWatch journaux détectent qu'une politique approche cette limite de taille, ils activent automatiquement les groupes de journaux commençant par`/aws/vendedlogs/`.

  Lorsque vous créez une règle d'alerte avec la journalisation activée, Amazon Managed Service for Prometheus doit mettre à jour CloudWatch votre politique de ressources de journaux avec le groupe de journaux que vous spécifiez. Pour éviter d'atteindre la limite de taille de la politique de gestion des CloudWatch journaux, préfixez les noms de vos groupes de CloudWatch journaux par`/aws/vendedlogs/`. Lorsque vous créez un groupe de journaux dans la console Amazon Managed Service for Prometheus, les noms des groupes de journaux ont le préfixe `/aws/vendedlogs/`. Pour plus d'informations, consultez la section Activation de la journalisation à partir de certains AWS services dans le guide de l'utilisateur CloudWatch des journaux.

Gestion du coût des requêtes dans Amazon Managed Service for Prometheus

Amazon Managed Service for Prometheus permet de limiter le coût des requêtes en limitant la quantité d'échantillons de requêtes traités (QSP) pouvant être utilisés par une seule requête. Vous

pouvez configurer deux types de seuils pour le QSP, les seuils d'avertissement et d'erreur, afin de gérer et de contrôler efficacement les coûts des requêtes.

Lorsque les requêtes atteignent le seuil d'avertissement, un message d'avertissement apparaît dans la réponse à la requête de l'API. Pour les requêtes consultées via Amazon Managed Grafana, l'avertissement sera visible dans l'interface utilisateur d'Amazon Managed Grafana, ce qui aidera les utilisateurs à identifier les requêtes coûteuses. Les requêtes qui atteignent le seuil d'erreur ne sont pas facturées et seront rejetées en cas d'erreur.

Outre la limitation des requêtes, Amazon Managed Service for Prometheus permet d'enregistrer les données de performance des requêtes dans Logs. CloudWatch Cette fonctionnalité vous permet d'analyser les requêtes en détail, ce qui vous aide à optimiser votre Amazon Managed Service pour les requêtes Prometheus et à gérer les coûts de manière plus efficace. La journalisation des requêtes capture des informations sur les requêtes qui dépassent les seuils QSP (Query Samples Processed) spécifiés. Ces données sont ensuite publiées dans CloudWatch Logs, ce qui vous permet d'étudier et d'analyser les performances des requêtes. Les requêtes enregistrées incluent à la fois les requêtes d'API et les requêtes de règles. Par défaut, la journalisation des requêtes est désactivée afin de minimiser l'utilisation inutile CloudWatch des journaux. Vous pouvez activer cette fonctionnalité lorsque cela est nécessaire pour l'analyse des requêtes.

Rubriques

- Configuration de la journalisation des requêtes
- Configuration des seuils de limitation des requêtes
- Contenu du journal
- Limitations

Configuration de la journalisation des requêtes

Vous pouvez configurer la journalisation des requêtes dans la console Amazon Managed Service for Prometheus ou dans la AWS CLI en appelant `create-query-logging-configuration` la demande d'API. Ce corps d'API contient une liste de destinations, mais pour le moment, nous ne prenons en charge que CloudWatch les journaux en tant que destination et les destinations doivent contenir exactement un élément avec des CloudWatch configurations.

Conditions préalables

Assurez-vous que le `logGroup` est déjà créé. L'ID ou le rôle utilisé pour la configuration doit être soumis à la politique suivante ou à des autorisations équivalentes.

JSON

```
{
"Version":"2012-10-17",
    "Statement": [
        {
"Effect": "Allow",
            "Action": [
                "logs:CreateLogDelivery",
                "logs:GetLogDelivery",
                "logs:UpdateLogDelivery",
                "logs:DeleteLogDelivery",
                "logs:ListLogDeliveries",
                "logs:PutResourcePolicy",
                "logs:DescribeResourcePolicies",
                "logs:DescribeLogGroups",
                "aps:CreateQueryLoggingConfiguration",
                "aps:UpdateQueryLoggingConfiguration",
                "aps:DescribeQueryLoggingConfiguration",
                "aps:DeleteQueryLoggingConfiguration"
            ],
            "Resource": "*"
        }
    ]
}
```

Configurer les CloudWatch journaux

Vous pouvez configurer les CloudWatch journaux en vous connectant à Amazon Managed Service for Prometheus à l'aide du ou Console de gestion AWS du. AWS CLI

Pour configurer la journalisation des requêtes à l'aide de la console Amazon Managed Service pour Prometheus

1. Accédez à l'onglet Journaux dans le volet des détails de votre espace de travail.

2. Sous Query Insights, choisissez Create.
3. Sélectionnez le menu déroulant Groupe de journaux et choisissez le groupe de journaux dans lequel publier vos journaux.

   Vous pouvez également créer un nouveau groupe de journaux dans la CloudWatch console.
4. Entrez le seuil (QSP).
5. Choisissez Enregistrer.

Pour configurer la journalisation des requêtes à l' AWS CLI aide de la commande

```
aws amp create-query-logging-configuration \
--workspace-id my_workspace_ID \
--destinations '[{"cloudWatchLogs":{"logGroupArn":"$my-log-group-arn"},"filters":
{"qspThreshold":$qspThreshold}}]'
```

Pour plus d'informations sur la mise à jour, la suppression et la description des opérations, consultez le manuel de référence de l'API Amazon Managed Service for Prometheus.

Configuration des seuils de limitation des requêtes

Pour configurer les seuils QSP, vous devez fournir les paramètres de requête dans l'QueryMetrics API.

- max_samples_processed_warning_threshold — Définit le seuil d'avertissement pour les échantillons de requêtes traités
- max_samples_processed_error_threshold — Définit le seuil d'erreur pour les échantillons de requêtes traités

Pour les utilisateurs d'Amazon Managed Grafana, vous pouvez utiliser la configuration de la source de données grafana pour appliquer des limites à toutes les requêtes provenant de la source de données :

1. Accédez à la configuration de la source de données Amazon Managed Service for Prometheus dans Amazon Managed Grafana.
2. Sous Paramètres de requête personnalisés, ajoutez les en-têtes de seuil.
3. Choisissez Enregistrer.

Contenu du journal

Pour les requêtes qui proviennent de règles, les informations suivantes relatives à la requête sont affichées dans les CloudWatch journaux :

```
{
    workspaceId: "workspace_id",
    message: {
        query: "avg(rate(go_goroutines[1m])) > 1",
        name: "alert_rule",
        kind: "alerting",
        group: "test-alert",
        namespace: "test",
        samples: "59321",
    },
    component: "ruler"
}
```

Pour les requêtes provenant d'appels d'API, vous verrez les informations suivantes concernant la requête dans les CloudWatch journaux :

```
{
    workspaceId: "ws-5e7658c2-7ccf-4c30-9de9-2ab26fa30639",
    message: {
        query: "sum by (instance) (go_memstats_alloc_bytes{job=\"node\"})",
        queryType: "range",
        start: "1683308700000",
        end: "1683913500000",
        step: "300000",
        samples: "11496",
        userAgent: "AWSPrometheusDPJavaClient/2.0.436.0 ",
        dashboardUid: "11234",
        panelId: "12"
    },
    component: "query-frontend"
}
```

Limitations

Limites de taille des politiques — Les politiques relatives aux ressources des CloudWatch journaux sont limitées à 5 120 caractères. Lorsque CloudWatch Logs détecte que la politique approche

de la limite de taille, il active automatiquement les groupes de journaux commençant par`/aws/vendedlogs/`. Lorsque vous activez la journalisation des requêtes, Amazon Managed Service for Prometheus doit mettre à jour CloudWatch votre politique de ressources de journaux avec le groupe de journaux que vous spécifiez. Pour éviter d'atteindre la limite de taille de la politique de gestion des CloudWatch journaux, préfixez les noms de vos groupes de CloudWatch journaux par`/aws/vendedlogs/`.

Comprenez et optimisez les coûts dans Amazon Managed Service for Prometheus

Les questions fréquemment posées ci-dessous et leurs réponses peuvent être utiles pour comprendre et optimiser les coûts associés à Amazon Managed Service for Prometheus.

Qu'est-ce qui contribue à mes coûts ?

Pour la plupart des clients, l'ingestion de métriques représente la majeure partie des coûts. Les clients qui utilisent beaucoup de requêtes ont également des coûts liés au traitement des échantillons de requêtes ; le stockage des métriques ne représentant qu'une faible part des coûts globaux. Pour plus d'informations sur les prix de chacun de ces éléments, consultez la section Tarification sur la page Amazon Managed Service for Prometheus.

Quel est le meilleur moyen de réduire mes coûts ? Comment réduire les coûts d'ingestion ?

Les taux d'ingestion (et non le stockage des métriques) constituent la majeure partie des coûts pour la plupart des clients. Vous pouvez réduire les taux d'ingestion en réduisant la fréquence de collecte (en augmentant l'intervalle de collecte) ou en réduisant le nombre de séries actives ingérées.

Vous pouvez augmenter l'intervalle de collecte (scraping) depuis votre agent de collecte : le serveur Prometheus (exécuté en mode Agent) et AWS le collecteur Distro OpenTelemetry for (ADOT) prennent en charge la configuration. `scrape_interval` Par exemple, l'augmentation de l'intervalle de collecte de 30 à 60 secondes réduira de moitié votre consommation d'ingestion.

Vous pouvez également filtrer les métriques envoyées à Amazon Managed Service for Prometheus à l'aide de `<relabel_config>`. Pour plus d'informations sur le réétiquetage dans la configuration de l'agent Prometheus, consultez #relabel_config https://prometheus.io/docs/prometheus/latest/configuration/configuration/ dans la documentation Prometheus.

Quel est le meilleur moyen de réduire mes coûts de requête ?

Les coûts des requêtes sont basés sur le nombre d'échantillons traités. Vous pouvez réduire la fréquence des requêtes afin de réduire les coûts liés aux requêtes.

Pour obtenir une meilleure visibilité sur les requêtes qui contribuent le plus aux coûts de vos requêtes, consultezGestion du coût des requêtes dans Amazon Managed Service for Prometheus.

Si je réduis la période de conservation de mes métriques, cela contribuera-t-il à réduire ma facture totale ?

Vous pouvez réduire votre période de conservation, mais il est peu probable que cela réduise considérablement vos coûts.

Pour plus d'informations sur la configuration de la période de rétention d'un espace de travail, consultezConfigurez votre espace de travail.

Comment puis-je réduire le coût de mes requêtes d'alerte ?

Les alertes créent des requêtes portant sur vos données, ce qui augmente les coûts de vos requêtes. Voici quelques stratégies que vous pouvez utiliser pour optimiser vos requêtes d'alerte et réduire vos coûts.

- Utiliser Amazon Managed Service pour les alertes Prometheus : les systèmes d'alerte externes à Amazon Managed Service for Prometheus peuvent nécessiter des requêtes supplémentaires pour renforcer la résilience ou la haute disponibilité, car le service externe interroge les métriques provenant de plusieurs zones de disponibilité ou régions. Cela inclut les alertes dans Grafana pour une haute disponibilité. Cela peut multiplier vos coûts par trois ou plus. Les alertes d'Amazon Managed Service for Prometheus sont optimisées et vous garantissent une disponibilité et une résilience élevées avec le plus petit nombre de requêtes possible.

  Nous vous recommandons d'utiliser les alertes natives dans Amazon Managed Service for Prometheus plutôt que des systèmes d'alerte externes.
- Optimisez votre intervalle d'alerte — Un moyen rapide d'optimiser vos requêtes d'alerte consiste à augmenter l'intervalle d'actualisation automatique. Si une alerte émet des requêtes toutes les minutes, mais qu'elle n'est nécessaire que toutes les cinq minutes, l'augmentation de l'intervalle d'actualisation automatique peut vous faire économiser cinq fois le coût des requêtes associées à cette alerte.
- Utilisez un effet rétrospectif optimal : une fenêtre de rétrospective plus grande dans votre requête augmente le coût de celle-ci, car elle extrait davantage de données. Assurez-vous que la fenêtre rétrospective de votre requête ProMQL est de taille raisonnable pour les données que vous devez

alerter. Par exemple, dans la règle suivante, l'expression inclut une fenêtre rétrospective de dix minutes :

```
- alert: metric:alerting_rule
  expr: avg(rate(container_cpu_usage_seconds_total[10m])) > 0
  for: 2m
```

La modification de la `expr` valeur à `avg(rate(container_cpu_usage_seconds_total[5m])) > 0` peut vous aider à réduire les coûts de vos requêtes.

En général, examinez vos règles d'alerte et assurez-vous que les alertes sont basées sur les meilleurs indicateurs pour votre service. Il est facile de créer des alertes qui se chevauchent sur les mêmes indicateurs ou plusieurs alertes qui vous fournissent les mêmes informations, en particulier lorsque vous ajoutez des alertes au fil du temps. Si vous constatez que vous voyez souvent des groupes d'alertes se produire en même temps, il est possible que vous puissiez optimiser vos alertes et ne pas les inclure toutes.

Ces suggestions peuvent vous aider à réduire les coûts. En fin de compte, vous devez équilibrer les coûts tout en créant le bon ensemble d'alertes pour comprendre l'état de votre système.

Pour plus d'informations sur les alertes dans Amazon Managed Service for Prometheus, consultez. Gestion et transfert des alertes dans Amazon Managed Service for Prometheus avec le gestionnaire d'alertes

Puis-je consulter ma facture à tout moment ?

Il AWS Cost and Usage Report suit votre AWS utilisation et fournit une estimation des frais associés à votre compte au cours d'une période de facturation. Pour plus d'informations, voir Que sont les rapports de AWS coûts et d'utilisation ? dans le guide de l'utilisateur des rapports sur les AWS coûts et l'utilisation

Quelles métriques puis-je utiliser pour surveiller mes coûts ?

Les échantillons métriques que vous ingérez constituent le principal facteur de coût d'Amazon Managed Service for Prometheus. Le nombre d'échantillons ingérés détermine directement vos frais mensuels. Il est donc essentiel de surveiller et de comprendre vos habitudes d'ingestion.

AWS Cost Explorerest la source fiable pour surveiller les coûts liés à Prometheus sur votre Amazon Managed Service. Vous pouvez suivre Cost Explorer pour connaître l'historique et les tendances quotidiennes des coûts sur Amazon Managed Service for Prometheus sur plusieurs aspects, y compris les échantillons ingérés. AWS La détection des anomalies de coûts peut également vous permettre de surveiller les changements inattendus de vos habitudes de dépenses.

L'utilisation de `IngestionRate` métriques fournit une méthode auxiliaire pour surveiller les tendances en matière d'ingestion qui sont directement corrélées aux coûts. Les avantages de l'utilisation en `IngestionRate` tant que métrique supplémentaire incluent :

- Workspace-level suivi — Surveillez l'ingestion par espace de travail plutôt qu'au niveau du compte uniquement.
- Visibilité granulaire : suivez les modèles d'ingestion toutes les heures, voire minute par minute, pour obtenir des informations en temps réel.
- Surveillance proactive : configurez des CloudWatch alarmes pour détecter les pics d'utilisation avant qu'ils n'apparaissent dans la facturation.

> **Note**
>
> `IngestionRate`peut être utilisé pour estimer les coûts et les tendances ou pour attribuer le coût par espace de travail, mais il n'est pas précis à 100 %. Comme `IngestionRate` les rapports indiquent un taux moyen échantillonné à intervalles d'une minute, la multiplication de ce taux par le temps fournit une approximation plutôt qu'un nombre exact d'échantillons ingérés. En outre, la politique CloudWatch de conservation des données d'Amazon affecte la granularité disponible pour les requêtes historiques, les données datant de plus de 63 jours étant limitées à des intervalles d'une heure.

Pour plus d'informations sur la surveillance des métriques CloudWatch Amazon Managed Service for Prometheus dans, consultez. Utilisez CloudWatch des métriques pour surveiller les ressources Amazon Managed Service for Prometheus

Surveillance des coûts des histogrammes natifs — Les histogrammes natifs sont mesurés sur la base de compartiments remplis : chaque compartiment rempli (un compartiment contenant des observations différentes de zéro) compte pour 0,25 d'un échantillon métrique, et les compartiments vides ne sont pas mesurés. Par exemple, un histogramme natif comportant 10 compartiments remplis est mesuré sous forme de 2,5 échantillons métriques. Pour surveiller les inducteurs de coûts

de votre histogramme natif, vous pouvez utiliser la `NativeHistogramIngestedBucketsRate` métrique pour surveiller le taux d'ingestion de compartiments remplis. Vous pouvez également l'utiliser `NativeHistogramIngestionRate` pour suivre le taux global d'ingestion d'échantillons d'histogrammes natifs et `NativeHistogramActiveSeries` pour surveiller le nombre de séries d'histogrammes natifs actives dans votre espace de travail.

Comment puis-je consulter mes coûts dans AWS Cost Explorer?

En tant que source fiable pour les coûts d'Amazon Managed Service for Prometheus AWS Cost Explorer , fournit votre utilisation réelle facturée et les frais d'Amazon Managed Service for Prometheus, les échantillons ingérés, y compris les données de facturation historiques par mois et par région. Utilisez Cost Explorer pour connaître le montant final de vos factures et l'évolution des coûts au jour le jour.

Pour consulter les coûts de votre Amazon Managed Service for Prometheus :

Accès AWS Cost Explorer

1. Connectez-vous à la console AWS de gestion.
2. Accédez au tableau de bord Billing and Cost Management.
3. Sélectionnez Cost Explorer dans le menu de navigation de gauche.
4. Choisissez Launch Cost Explorer (si c'est la première fois que vous l'utilisez).

Configuration du rapport

1. Définissez votre plage horaire en fonction de la période de facturation souhaitée (par exemple, mars 2025 - février 2026).
2. Sous Filtres, sélectionnez :
   - Service : Choisissez « Amazon Managed Service for Prometheus ».
   - Type d'utilisation : Filtrez pour « MetricSampleCount » pour isoler les frais d'ingestion d'échantillons.

Regrouper et afficher les données

1. Sous Regrouper par, sélectionnez Région pour afficher les données de coût et d'utilisation par région.

2. Choisissez votre visualisation préférée (graphique à barres, graphique linéaire ou tableau).
3. Choisissez Appliquer pour générer le rapport.

Exporter les données (facultatif)

1. Choisissez Télécharger le fichier CSV dans le coin supérieur droit pour exporter les données.
2. Le fichier CSV contiendra : la période de facturation, la région, le type d'utilisation, le montant facturé et la quantité d'utilisation (nombre d'échantillons facturés).

> **Note**
>
> Les données de Cost Explorer sont généralement retardées de 24 heures. Pour la période de facturation la plus récente, les données peuvent ne pas être disponibles avant le jour suivant.

Comment calculer le nombre d'échantillons ingérés en un mois ?

Vous pouvez calculer le nombre approximatif d'échantillons ingérés à l'aide des `IngestionRate` statistiques CloudWatch d'Amazon avec le AWS Command Line Interface. Cela est utile pour consulter les factures mensuelles et comprendre les modèles d'utilisation dans les espaces de travail.

Pour récupérer les données d'ingestion :

```
aws cloudwatch get-metric-data \
    --region your-region \
    --start-time start-timestamp \
    --end-time end-timestamp \
    --metric-data-queries '[
        {
            "Id": "e1",
            "Expression": "SUM(METRICS())",
            "Period": 3600
        },
        {
            "Id": "ws1",
            "MetricStat": {
                "Metric": {
                    "Namespace": "AWS/Usage",
                    "MetricName": "ResourceCount",
```

```
                    "Dimensions": [
                        {"Name": "Service",    "Value": "Prometheus"},
                        {"Name": "Resource",   "Value": "IngestionRate"},
                        {"Name": "Type",       "Value": "Resource"},
                        {"Name": "Class",      "Value": "None"},
                        {"Name": "ResourceId", "Value": "YOUR_AMP_WORKSPACE_ID"}
                    ]
                },
                "Period": 3600,
                "Stat": "Average"
            }
        }
    ]'
```

La commande renvoie des `IngestionRate` valeurs moyennes horaires, mesurées en échantillons par seconde. Pour calculer le nombre approximatif d'échantillons ingérés au cours d'un mois, multipliez chaque point de données horaire par 3 600 (secondes par heure) pour obtenir les échantillons ingérés au cours de cette heure, puis additionnez tous les totaux horaires du mois :

```
Monthly samples ≈ Σ (hourly IngestionRate average × 3600)
```

Par exemple, si une heure renvoie en moyenne `IngestionRate` 500 échantillons par seconde, cette heure a contribué à environ 500 × 3 600 = 1 800 000 échantillons. Répétez cette opération toutes les heures du mois et additionnez les résultats pour obtenir le nombre approximatif de vos doses mensuelles.

Paramètres clés :

- `Period`: 3600 (1 heure en secondes)
- `StartTime`: Début de votre mois (par exemple,`2026-02-01T00:00:00Z`)
- `EndTime`: Fin de votre mois (par exemple,`2026-03-01T00:00:00Z`)
- `Stat`: Moyen

Pour trouver les identifiants de votre espace de travail :

```
aws amp list-workspaces --region your-region
```

Utilisez l'ID de l'espace de travail pour filtrer les statistiques afin d'afficher les données uniquement pour l'espace de travail spécifié, plutôt que de les agréger entre toutes les ressources Prometheus de la région.

Quelle granularité des données est disponible pour l'analyse des coûts historiques ?

La politique CloudWatch de conservation des données d'Amazon affecte la granularité disponible pour les requêtes historiques :

- Données datant de moins de 15 jours : requête à intervalles d'une minute (`Period`: 60)
- Données datant de 15 à 63 jours : requête à intervalles de 5 minutes (`Period`: 300)
- Données datant de plus de 63 jours : limitées à des intervalles d'une heure (`Period`: 3600)

Pour une analyse historique au-delà de 63 jours, CloudWatch sous-échantillonne automatiquement les données sur une période minimale d'une heure. Lorsque vous examinez la facturation pour des mois antérieurs à 63 jours, vous devez utiliser des données agrégées horaires. Le calcul de l'échantillon mensuel utilise ces points de données moyens horaires, en additionnant chaque valeur multipliée par 3 600 sur l'ensemble du mois.

Cette granularité réduite explique en outre pourquoi les données anciennes `IngestionRate` fournissent des estimations plutôt que des chiffres exacts. Reportez-vous toujours à Cost Explorer pour connaître les montants facturés faisant autorité.

Pour plus d'informations sur la rétention des CloudWatch métriques, consultez la section Conservation des métriques dans le guide de CloudWatch l'utilisateur Amazon.

Quelles sont les meilleures pratiques pour surveiller les coûts d'Amazon Managed Service for Prometheus ?

Pour gérer et optimiser efficacement vos dépenses liées à Amazon Managed Service for Prometheus, envisagez de mettre en œuvre les pratiques de surveillance suivantes :

- Surveillez régulièrement Cost Explorer pour suivre les tendances réelles des dépenses et identifier les anomalies de coûts dans de multiples dimensions, y compris les échantillons ingérés.

- Activez la détection des anomalies de AWS coûts pour recevoir des alertes en cas d'augmentation imprévue des coûts de votre Amazon Managed Service for Prometheus.
- Activez les CloudWatch alarmes `IngestionRate` pour la surveillance au niveau de l'espace de travail et la détection précoce des pics d'ingestion.
- Exportez régulièrement les données de Cost Explorer pour une analyse des coûts et des rapports à long terme.

Pourquoi ma facture est-elle plus élevée en début de mois qu'en fin de mois ?

Amazon Managed Service for Prometheus propose un modèle de tarification échelonné pour l'ingestion, ce qui se traduit par une augmentation des coûts liés à votre utilisation initiale. À mesure que votre consommation atteint des niveaux d'ingestion plus élevés, avec des coûts plus faibles, vos coûts diminuent. Pour plus d'informations sur la tarification, notamment les niveaux d'ingestion, consultez la section Tarification sur la page Amazon Managed Service for Prometheus.

> **Note**
>
> - Les niveaux sont destinés à être utilisés au sein d'une région, et non entre les régions. L'utilisation au sein d'une région doit atteindre le niveau suivant pour bénéficier du tarif inférieur.
> - Dans une organisation en AWS Organizations, l'utilisation des niveaux est comptabilisée par compte payeur, et non par compte (le compte payeur est toujours le compte de gestion de l'organisation). Lorsque le total des mesures ingérées (au sein d'une région) pour tous les comptes d'une organisation atteint le niveau suivant, le taux le plus bas est facturé à tous les comptes.

J'ai supprimé tous mes espaces de travail Amazon Managed Service for Prometheus, mais il semblerait que je sois toujours débité. Qu'est-ce qui pourrait se passer ?

Dans ce cas, il est possible que vous disposiez toujours de scrapers AWS gérés configurés pour envoyer des métriques à vos espaces de travail supprimés. Suivez les instructions pourRecherche et suppression des scrapers.

Intégration à d'autres AWS services

Amazon Managed Service for Prometheus s'intègre à d'autres services. AWS Cette section décrit l'intégration à la surveillance des coûts d'Amazon Elastic Kubernetes Service (Amazon EKS) (avec Kubecost) et explique comment intégrer des métriques à l'aide d'Amazon Data Firehose. CloudWatch Il décrit également la configuration et la gestion d'Amazon Managed Service pour Prometheus à l' AWS aide des modules Observability Accelerator Terraform ou à l'aide de contrôleurs pour Kubernetes. AWS

Rubriques

- Intégration au suivi des coûts Amazon EKS
- Configurer Amazon Managed Service pour AWS Prometheus avec Observability Accelerator
- Gérez Amazon Managed Service pour Prometheus AWS avec des contrôleurs pour Kubernetes
- Intégration CloudWatch des métriques à Amazon Managed Service pour Prometheus

Intégration au suivi des coûts Amazon EKS

Amazon Managed Service for Prometheus s'intègre au suivi des coûts d'Amazon Elastic Kubernetes Service (Amazon EKS) (avec Kubecost) pour effectuer des calculs de répartition des coûts et fournir des informations sur l'optimisation de vos clusters Kubernetes. L'utilisation d'Amazon Managed Service for Prometheus avec Kubecost vous permet d'adapter de manière fiable votre suivi des coûts pour prendre en charge des clusters plus importants.

L'intégration à Kubecost vous donne une visibilité précise sur les coûts de votre cluster Amazon EKS. Vous pouvez agréger les coûts selon la majorité des contextes Kubernetes, du niveau du conteneur jusqu'au niveau du cluster, et même le niveau de plusieurs clusters. Vous pouvez générer des rapports sur l'ensemble des conteneurs ou des clusters afin de suivre les coûts à des fins de démonstration ou de rétrofacturation.

Vous trouverez ci-dessous des instructions pour l'intégration à Kubecost dans un scénario à un ou plusieurs clusters :

- Intégration à un seul cluster – Pour savoir comment intégrer le suivi des coûts d'Amazon EKS à un seul cluster, consultez l'article du blog AWS Integrating Kubecost with Amazon Managed Service for Prometheus.

- Intégration à plusieurs clusters – Pour savoir comment intégrer le suivi des coûts d'Amazon EKS à plusieurs clusters, consultez l'article du blog AWS Multi-cluster cost monitoring for Amazon EKS using Kubecost and Amazon Managed Service for Prometheus.

> **Note**
>
> Pour plus d'informations sur l'utilisation de Kubecost, consultez la section Suivi des coûts dans le Guide de l'utilisateur Amazon EKS.

Configurer Amazon Managed Service pour AWS Prometheus avec Observability Accelerator

AWS fournit des outils d'observabilité, notamment la surveillance, la journalisation, les alertes et les tableaux de bord, pour vos projets Amazon Elastic Kubernetes Service (Amazon EKS). Cela inclut Amazon Managed Service pour Prometheus, Amazon Managed AWS Grafana, OpenTelemetry Distro for et d'autres outils. Pour vous aider à utiliser ces outils ensemble, AWS propose des modules Terraform qui configurent l'observabilité avec ces services, appelés Accélérateur d'observabilité.AWS

AWS Observability Accelerator fournit deux profils de collecteur pour Amazon Managed Service for Prometheus :

- Métriques gérées (sans agent) : utilise le service géré Amazon pour Prometheus Collector, un scraper entièrement géré et sans agent qui s'exécute en dehors de votre cluster. Aucun module de collecte à gérer. Métriques uniquement.
- Autogéré — Déploie un OpenTelemetry collecteur via Helm dans votre cluster. Supporte les métriques, les traces (AWS X-Ray) et les journaux (Amazon CloudWatch).

Cette section décrit les deux options, en commençant par l'approche sans agent recommandée.

Les modèles Terraform et les instructions détaillées se trouvent sur la page AWS Observability Accelerator for Terraform. GitHub

Conditions préalables

Pour utiliser AWS Observability Accelerator, vous devez disposer d'un cluster Amazon EKS existant et remplir les conditions préalables suivantes :

- AWS CLI— utilisé pour appeler une AWS fonctionnalité depuis la ligne de commande.
- kubectl : utilisé pour contrôler votre cluster EKS à partir de la ligne de commande.
- Terraform (>= 1.5.0) : utilisé pour automatiser la création des ressources pour cette solution. Vous devez configurer le AWS fournisseur avec un rôle IAM autorisé à créer et à gérer Amazon Managed Service pour Prometheus, Amazon Managed Grafana et IAM au sein de votre compte. AWS Pour plus d'informations sur la configuration du fournisseur AWS pour Terraform, consultez la section AWS provider dans la documentation Terraform.

Utilisation de l'exemple de métriques gérées (sans agent)

Cet exemple utilise le collecteur Amazon Managed Service for Prometheus pour extraire les métriques Prometheus de votre cluster Amazon EKS sans déployer de pods de collecte. Le collecteur nécessite au moins deux sous-réseaux dans deux zones de disponibilité distinctes. Pour plus de détails, consultez l'eks-amp-managedexemple sur GitHub.

Pour utiliser le module Terraform de surveillance de l'infrastructure sans agent

1. Dans le dossier dans lequel vous souhaitez créer votre projet, clonez le référentiel à l'aide de la commande suivante.

```
git clone https://github.com/aws-observability/terraform-aws-observability-
accelerator.git
```

2. Initialisez Terraform avec les commandes suivantes.

```
cd examples/eks-amp-managed

terraform init
```

3. Créez un nouveau fichier `terraform.tfvars`, comme dans l'exemple suivant. Utilisez les informations relatives à AWS la région, à l'ID du cluster et au réseau VPC de votre cluster Amazon EKS. Le collecteur nécessite au moins deux sous-réseaux dans deux zones de disponibilité distinctes.

```
# (mandatory) AWS Region where your resources will be located
aws_region = "eu-west-1"

# (mandatory) EKS Cluster name
eks_cluster_id = "my-eks-cluster"
```

```
# (mandatory) Subnets for the managed scraper (>= 2 AZs)
scraper_subnet_ids = ["subnet-aaa", "subnet-bbb"]

# (mandatory) Security group allowing scraper access to the EKS API
scraper_security_group_ids = ["sg-xxx"]
```

4. Créez un espace de travail Amazon Managed Grafana, si vous n'en avez pas déjà. Pour plus d'informations sur la création d'un nouvel espace de travail, consultez la section Create your first workspace dans le Guide de l'utilisateur Amazon Managed Grafana.
5. Créez deux variables pour que Terraform utilise votre espace de travail Grafana en exécutant les commandes suivantes sur la ligne de commande. Vous devrez le remplacer par l'identifiant *grafana-workspace-id* de votre espace de travail Grafana.

```
export TF_VAR_managed_grafana_workspace_id=grafana-workspace-id
export TF_VAR_grafana_api_key=`aws grafana create-workspace-api-key --key-name
 "observability-accelerator-$(date +%s)" --key-role ADMIN --seconds-to-live 1200 --
workspace-id $TF_VAR_managed_grafana_workspace_id --query key --output text`
```

6. [Facultatif] Pour utiliser un espace de travail Amazon Managed Service for Prometheus existant, ajoutez l'identifiant au fichier, comme dans `terraform.tfvars` l'exemple suivant, en le remplaçant *prometheus-workspace-id* par votre identifiant d'espace de travail Prometheus. Si vous ne spécifiez pas d'espace de travail existant, un nouvel espace de travail Prometheus sera créé pour vous.

```
# (optional) Leave it empty for a new workspace to be created
managed_prometheus_workspace_id = "prometheus-workspace-id"
```

7. Déployez la solution à l'aide de la commande suivante.

```
terraform apply -var-file=terraform.tfvars
```

Cela créera des ressources dans votre AWS compte, notamment les suivantes :

- Un nouvel espace de travail Amazon Managed Service for Prometheus (sauf si vous avez choisi d'utiliser un espace de travail existant).
- Un collecteur Amazon Managed Service for Prometheus (scraper sans agent) configuré pour extraire les métriques Prometheus de votre cluster Amazon EKS.

- Règles d'enregistrement et d'alerte Prometheus dans votre espace de travail Amazon Managed Service for Prometheus.
- kube-state-metrics et un exportateur de nœuds déployé dans votre cluster Amazon EKS pour les métriques d'infrastructure.
- Une nouvelle source de données et de nouveaux tableaux de bord Amazon Managed Grafana dans votre espace de travail actuel. Les tableaux de bord seront répertoriés sous EKS Monitoring.

Alternative : collecteur autogéré OpenTelemetry

Si vous avez besoin de traces, de journaux ou d'un contrôle total sur le pipeline de collecte, utilisez le profil autogéré. Cela déploie un OpenTelemetry collecteur via Helm dans votre cluster Amazon EKS, configuré pour extraire les métriques Prometheus et écrire à distance sur Amazon Managed Service for Prometheus. Il prend également en charge les traces (AWS X-Ray) et les journaux (Amazon CloudWatch). Pour plus de détails, consultez l'eks-amp-otelexemple sur GitHub.

Pour utiliser le module Terraform autogéré

1. Clonez le dépôt et initialisez Terraform.

```
git clone https://github.com/aws-observability/terraform-aws-observability-
accelerator.git
cd examples/eks-amp-otel
terraform init
```

2. Créez un nouveau fichier `terraform.tfvars`, comme dans l'exemple suivant.

```
# (mandatory) AWS Region where your resources will be located
aws_region = "eu-west-1"

# (mandatory) EKS Cluster name
eks_cluster_id = "my-eks-cluster"
```

3. Configurez votre espace de travail Amazon Managed Grafana et votre clé d'API en suivant les mêmes étapes que dans l'exemple des métriques gérées (étapes 4 à 6 ci-dessus).
4. Déployez la solution à l'aide de la commande suivante.

```
terraform apply -var-file=terraform.tfvars
```

Cela créera les ressources suivantes dans votre AWS compte (contrairement à l'approche sans agent, le collecteur s'exécute au sein de votre cluster) :

- Un espace de travail Amazon Managed Service pour Prometheus (s'il n'est pas fourni).
- Un espace de travail Grafana géré par Amazon avec source de données et tableaux de bord.
- Un OpenTelemetry collecteur déployé via Helm dans votre cluster Amazon EKS, configuré pour extraire les métriques Prometheus et écrire à distance sur Amazon Managed Service for Prometheus.
- Rôle IAM pour les comptes de service (IRSA) pour le OpenTelemetry Collector.
- Trace le pipeline jusqu'à AWS X-Ray (activé par défaut).
- Logs le pipeline vers Amazon CloudWatch (activé par défaut).

Affichage des tableaux de bord

Pour consulter vos nouveaux tableaux de bord, ouvrez le tableau de bord spécifique dans votre espace de travail Amazon Managed Grafana. Les tableaux de bord de l'infrastructure sont fournis automatiquement par Terraform. Pour plus d'informations sur l'utilisation d'Amazon Managed Grafana, consultez la section Working in your Grafana workspace dans le Guide de l'utilisateur Amazon Managed Grafana.

Gérez Amazon Managed Service pour Prometheus AWS avec des contrôleurs pour Kubernetes

Amazon Managed Service for Prometheus est intégré à AWS Controllers for Kubernetes (ACK) et permet de gérer votre espace de travail, le gestionnaire d'alertes et les ressources de règles dans Amazon EKS. Vous pouvez utiliser les définitions de ressources personnalisées AWS Controllers for Kubernetes (CRDs) et les objets Kubernetes natifs sans avoir à définir de ressources en dehors de votre cluster.

Cette section explique comment configurer les AWS contrôleurs pour Kubernetes et Amazon Managed Service pour Prometheus dans un cluster Amazon EKS existant.

Vous pouvez également lire les articles de blog présentant les AWS contrôleurs pour Kubernetes et le contrôleur ACK pour Amazon Managed Service for Prometheus.

Conditions préalables

Avant de commencer à intégrer AWS Controllers for Kubernetes et Amazon Managed Service for Prometheus à votre cluster Amazon EKS, vous devez remplir les conditions préalables suivantes.

- Vous devez disposer d'une autorisation Compte AWS et d'une autorisation pour créer des rôles Amazon Managed Service for Prometheus et IAM par programmation.
- Vous devez disposer d'un cluster Amazon EKS avec OpenID Connect (OIDC) activé.

  Si OIDC n'est pas activé, vous pouvez utiliser la commande suivante pour le faire. N'oubliez pas de remplacer le *YOUR_CLUSTER_NAME* et *AWS_REGION* par les valeurs correctes pour votre compte.

  ```
  eksctl utils associate-iam-oidc-provider \
      --cluster ${YOUR_CLUSTER_NAME} --region ${AWS_REGION} \
      --approve
  ```

  Pour plus d'informations sur l'utilisation d'OIDC avec Amazon EKS, consultez les sections Authentification du fournisseur d'identité OIDC et Creating an IAM OIDC provider dans le Guide l'utilisateur Amazon EKS.
- Le pilote CIS Amazon EBS doit être installé sur votre cluster Amazon EKS.
- L'AWS CLI doit être installée. Le AWS CLI est utilisé pour appeler des AWS fonctionnalités depuis la ligne de commande.
- Helm, le gestionnaire de packages pour Kubernetes, doit être installé.
- Les métriques du plan de contrôle avec Prometheus doivent être configurées dans votre cluster Amazon EKS.
- Vous devez disposer d'une rubrique Amazon Simple Notification Service (Amazon SNS) dans laquelle vous souhaitez envoyer des alertes à partir de votre nouvel espace de travail. Assurez-vous d'avoir autorisé Amazon Managed Service for Prometheus à envoyer des messages à la rubrique.

Lorsque votre cluster Amazon EKS est correctement configuré, vous devez être en mesure de voir les métriques formatées pour Prometheus en appelant `kubectl get --raw /metrics`. Vous êtes maintenant prêt à installer un contrôleur de service AWS Controllers for Kubernetes et à l'utiliser pour déployer les ressources Amazon Managed Service for Prometheus.

Déploiement d'un espace de travail avec AWS Controllers for Kubernetes

Pour déployer un nouvel espace de travail Amazon Managed Service pour Prometheus, vous devez installer AWS un contrôleur Controllers for Kubernetes, puis l'utiliser pour créer l'espace de travail.

Pour déployer un nouvel espace AWS de travail Amazon Managed Service pour Prometheus avec Controllers for Kubernetes

1. Les commandes suivantes permettent d'utiliser Helm pour installer le contrôleur de service Amazon Managed Service for Prometheus. Pour plus d'informations, consultez Installer un contrôleur ACK dans la documentation des AWS contrôleurs pour Kubernetes sur. GitHub Utilisez la version *region* adaptée à votre système, par exemple`us-east-1`.

   ```
   export SERVICE=prometheusservice
   export RELEASE_VERSION=`curl -sL https://api.github.com/repos/aws-controllers-k8s/
   $SERVICE-controller/releases/latest | jq -r '.tag_name | ltrimstr("v")'`
   export ACK_SYSTEM_NAMESPACE=ack-system
   export AWS_REGION=region

   aws ecr-public get-login-password --region us-east-1 | helm registry login --
   username AWS --password-stdin public.ecr.aws
   helm install --create-namespace -n $ACK_SYSTEM_NAMESPACE ack-$SERVICE-controller \
     oci://public.ecr.aws/aws-controllers-k8s/$SERVICE-chart --version=
   $RELEASE_VERSION --set=aws.region=$AWS_REGION
   ```

   Après quelques instants, vous devriez voir une réponse similaire à la suivante, qui indique la réussite de l'opération.

   ```
   You are now able to create Amazon Managed Service for Prometheus (AMP) resources!
   The controller is running in "cluster" mode.
   The controller is configured to manage AWS resources in region: "us-east-1"
   ```

   Vous pouvez éventuellement vérifier que le contrôleur AWS Controllers for Kubernetes a été correctement installé à l'aide de la commande suivante.

   ```
   helm list --namespace $ACK_SYSTEM_NAMESPACE -o yaml
   ```

   Cela renverra des informations sur le contrôleur `ack-prometheusservice-controller`, notamment `status: deployed`.

2. Créez un fichier appelé `workspace.yaml` avec le texte suivant. Il sera utilisé comme configuration pour l'espace de travail que vous créez.

```
apiVersion: prometheusservice.services.k8s.aws/v1alpha1
kind: Workspace
metadata:
  name: my-amp-workspace
spec:
  alias: my-amp-workspace
  tags:
    ClusterName: EKS-demo
```

3. Exécutez la commande suivante pour créer votre espace de travail (cette commande dépend des variables système que vous avez définies à l'étape 1).

```
kubectl apply -f workspace.yaml -n $ACK_SYSTEM_NAMESPACE
```

Après quelques instants, vous devriez être en mesure de voir un nouvel espace de travail appelé `my-amp-workspace` dans votre compte.

Exécutez la commande suivante pour afficher les détails et le statut de votre espace de travail, notamment l'ID de l'espace de travail. Vous pouvez également consulter le nouvel espace de travail dans la console Amazon Managed Service for Prometheus.

```
kubectl describe workspace my-amp-workspace -n $ACK_SYSTEM_NAMESPACE
```

> **Note**
>
> Vous pouvez également utiliser un espace de travail existant plutôt que d'en créer un.

4. Créez deux nouveaux fichiers yaml comme configuration pour les groupes de règles et AlertManager que vous créerez ensuite en utilisant la configuration suivante.

Enregistrez cette configuration sous `rulegroup.yaml`. *`WORKSPACE-ID`*Remplacez-le par l'ID de l'espace de travail de l'étape précédente.

```
apiVersion: prometheusservice.services.k8s.aws/v1alpha1
kind: RuleGroupsNamespace
metadata:
  name: default-rule
```

```
spec:
  workspaceID: WORKSPACE-ID
  name: default-rule
  configuration: |
    groups:
    - name: example
      rules:
      - alert: HostHighCpuLoad
        expr: 100 - (avg(rate(node_cpu_seconds_total{mode="idle"}[2m])) * 100) > 60
        for: 5m
        labels:
          severity: warning
          event_type: scale_up
        annotations:
          summary: Host high CPU load (instance {{ $labels.instance }})
          description: "CPU load is > 60%\n  VALUE = {{ $value }}\n  LABELS =
 {{ $labels }}"
      - alert: HostLowCpuLoad
        expr: 100 - (avg(rate(node_cpu_seconds_total{mode="idle"}[2m])) * 100) < 30
        for: 5m
        labels:
          severity: warning
          event_type: scale_down
        annotations:
          summary: Host low CPU load (instance {{ $labels.instance }})
          description: "CPU load is < 30%\n  VALUE = {{ $value }}\n  LABELS =
 {{ $labels }}"
```

Enregistrez la configuration suivante sous `alertmanager.yaml`. *WORKSPACE-ID*Remplacez-le par l'ID de l'espace de travail de l'étape précédente. *TOPIC-ARN*Remplacez-le par l'ARN de la rubrique Amazon SNS à laquelle envoyer des notifications, et *REGION* par le Région AWS que vous utilisez. N'oubliez pas qu'Amazon Managed Service for Prometheus doit disposer d'autorisations pour accéder à la rubrique Amazon SNS.

```
apiVersion: prometheusservice.services.k8s.aws/v1alpha1
kind: AlertManagerDefinition
metadata:
  name: alert-manager
spec:
  workspaceID: WORKSPACE-ID
  configuration: |
    alertmanager_config: |
```

```
    route:
       receiver: default_receiver
    receivers:
      - name: default_receiver
        sns_configs:
        - topic_arn: TOPIC-ARN
          sigv4:
            region: REGION
          message: |
            alert_type: {{ .CommonLabels.alertname }}
            event_type: {{ .CommonLabels.event_type }}
```

> **Note**
>
> Pour en savoir plus sur les formats de ces fichiers de configuration, consultez les sections RuleGroupsNamespaceData et AlertManagerDefinitionData.

5. Exécutez les commandes suivantes pour créer la configuration de votre groupe de règles et de votre gestionnaire d'alertes (cette commande dépend des variables système que vous avez définies à l'étape 1).

```
kubectl apply -f rulegroup.yaml -n $ACK_SYSTEM_NAMESPACE
kubectl apply -f alertmanager.yaml -n $ACK_SYSTEM_NAMESPACE
```

Les modifications seront disponibles en quelques instants.

> **Note**
>
> Pour mettre à jour une ressource, plutôt que de la créer, il suffit de mettre à jour le fichier yaml et de réexécuter la commande `kubectl apply`.
> Pour supprimer une ressource, exécutez la commande suivante. Remplacez *ResourceType* par le type de ressource que vous souhaitez supprimer `WorkspaceAlertManagerDefinition`, ou`RuleGroupNamespace`. Remplacez *ResourceName* par le nom de la ressource à supprimer.
>
> ```
> kubectl delete ResourceType ResourceName -n $ACK_SYSTEM_NAMESPACE
> ```

Le déploiement du nouvel espace de travail est terminé. La section suivante décrit la configuration de votre cluster pour envoyer des métriques à cet espace de travail.

Configuration de votre cluster Amazon EKS pour écrire dans l'espace de travail Amazon Managed Service for Prometheus

Cette section décrit comment utiliser Helm pour configurer l'instance Prometheus exécutée dans votre cluster Amazon EKS afin d'écrire à distance des métriques dans l'espace de travail Amazon Managed Service for Prometheus créé à la section précédente.

Pour cette procédure, vous aurez besoin du nom du rôle IAM que vous avez créé pour l'ingestion de métriques. Si vous ne l'avez pas déjà fait, consultez la section Configuration de rôles de service pour l'ingestion de métriques à partir de clusters Amazon EKS pour de plus amples informations et instructions. Si vous suivez ces instructions, le rôle IAM sera appelé `amp-iamproxy-ingest-role`.

Pour configurer votre cluster Amazon EKS pour l'écriture à distance

1. Utilisez la commande suivante pour obtenir le `prometheusEndpoint` pour votre espace de travail. *`WORKSPACE-ID`*Remplacez-le par l'ID de l'espace de travail indiqué dans la section précédente.

   ```
   aws amp describe-workspace --workspace-id WORKSPACE-ID
   ```

   Le point de terminaison prometheusEndpoint figurera dans les résultats renvoyés et sera formaté comme suit :

   ```
   https://aps-workspaces.us-west-2.amazonaws.com/workspaces/ws-a1b2c3d4-a123-b456-c789-ac1234567890/
   ```

   Enregistrez cette URL pour l'utiliser lors des prochaines étapes.

2. Créez un fichier avec le texte suivant et nommez-le `prometheus-config.yaml`. *`account`*Remplacez-le par votre identifiant de compte, *`workspaceURL/`* par l'URL que vous venez de trouver et *`region`* par l'URL appropriée Région AWS à votre système.

   ```
   serviceAccounts:
           server:
               name: "amp-iamproxy-ingest-service-account"
               annotations:
   ```

```
                eks.amazonaws.com/role-arn: "arn:aws:iam::account:role/amp-
iamproxy-ingest-role"
server:
    remoteWrite:
        - url: workspaceURL/api/v1/remote_write
          sigv4:
            region: region
          queue_config:
            max_samples_per_send: 1000
            max_shards: 200
            capacity: 2500
```

3. Recherchez le graphique Prometheus et les noms des espaces de noms ainsi que la version du graphique à l'aide de la commande Helm suivante.

```
helm ls --all-namespaces
```

   D'après les étapes effectuées jusqu'à présent, le graphique Prometheus et l'espace de noms doivent tous deux être nommés `prometheus`, et la version du graphique peut être `15.2.0`.

4. Exécutez la commande suivante en utilisant le *PrometheusChartNamePrometheusNamespace*, et *PrometheusChartVersion* trouvé à l'étape précédente.

```
helm upgrade PrometheusChartName prometheus-community/prometheus -
n PrometheusNamespace -f prometheus-config.yaml --version PrometheusChartVersion
```

   Après quelques minutes, un message s'affiche, indiquant que la mise à niveau a réussi.

5. Vous pouvez éventuellement vérifier que les métriques ont bien été envoyées en interrogeant le point de terminaison Amazon Managed Service for Prometheus via `awscurl`. *Region*Remplacez-le par celui Région AWS que vous utilisez et *workspaceURL/* par l'URL que vous avez trouvée à l'étape 1.

```
awscurl --service="aps" --region="Region" "workspaceURL/api/v1/query?
query=node_cpu_seconds_total"
```

Vous avez maintenant créé un espace de travail Amazon Managed Service for Prometheus et vous y êtes connecté depuis votre cluster Amazon EKS, en utilisant des fichiers YAML comme configuration. Ces fichiers, appelés définitions de ressources personnalisées (CRDs), se trouvent au sein de votre

cluster Amazon EKS. Vous pouvez utiliser le contrôleur AWS Controllers for Kubernetes pour gérer toutes vos ressources Amazon Managed Service for Prometheus directement depuis le cluster.

Intégration CloudWatch des métriques à Amazon Managed Service pour Prometheus

Il peut être utile d'avoir tous vos indicateurs au même endroit. Amazon Managed Service for Prometheus n'ingère pas automatiquement les métriques Amazon. CloudWatch Cependant, vous pouvez utiliser Amazon Data Firehose pour envoyer des CloudWatch métriques AWS Lambda à Amazon Managed Service for Prometheus.

Cette section explique comment instrumenter un flux de CloudWatch métriques Amazon, utiliser Amazon Data Firehose et AWS Lambdaintégrer des métriques dans Amazon Managed Service for Prometheus.

Vous allez configurer une pile à l'aide du AWS Cloud Development Kit (CDK) pour créer un Firehose Delivery Stream, un Lambda et un bucket Amazon S3 afin de présenter un scénario complet.

Infrastructure

La première chose à faire est de configurer l'infrastructure pour cette recette.

CloudWatch les flux métriques permettent de transférer les données métriques de streaming vers un point de terminaison HTTP ou un compartiment Amazon S3.

La mise en place de l'infrastructure se fait 4 étapes :

- Configuration des prérequis
- Création d'un espace de travail Amazon Managed Service for Prometheus
- Installation des dépendances
- Déploiement de la pile

Prérequis

- Le AWS CLI est installé et configuré dans votre environnement.
- Le TypeScript AWS CDK est installé dans votre environnement.
- Node.js et Go sont installés dans votre environnement.

- L'exportateur de CloudWatch métriques AWS d'observabilité github repository (`CWMetricsStreamExporter`) a été cloné sur votre machine locale.

Pour créer un espace de travail Amazon Managed Service for Prometheus

1. L'application de démonstration présentée dans cette recette sera exécutée sur Amazon Managed Service for Prometheus. Créez votre espace de travail Amazon Managed Service for Prometheus à l'aide de la commande suivante :

```
aws amp create-workspace --alias prometheus-demo-recipe
```

2. Assurez-vous que votre espace de travail a été créé à l'aide de la commande suivante :

```
aws amp list-workspaces
```

   Pour plus d'informations sur Amazon Managed Service for Prometheus consultez le Guide l'utilisateur Amazon Managed Service for Prometheus.

Pour installer les dépendances du kit

1. Installation des dépendances

   À la racine du référentiel `aws-o11y-recipes`, remplacez votre répertoire par `CWMetricStreamExporter` à l'aide de la commande suivante :

```
cd sandbox/CWMetricStreamExporter
```

   Ce sera désormais la racine du référentiel.

2. Remplacez le répertoire `/cdk` à l'aide de la commande suivante :

```
cd cdk
```

3. Installez les dépendances CDK à l'aide de la commande suivante :

```
npm install
```

4. Revenez au répertoire à la racine du référentiel, puis remplacez le répertoire par `/lambda` à l'aide de la commande suivante :

```
cd lambda
```

5. Dans le dossier `/lambda`, installez les dépendances Go en utilisant :

   ```
   go get
   ```

   Toutes les dépendances sont désormais installées.

Pour déployer la pile

1. À la racine du référentiel, ouvrez `config.yaml` et modifiez l'URL de l'espace de travail Amazon Managed Service for Prometheus en remplaçant `{workspace}` par l'ID du nouvel espace de travail, ainsi que la région dans laquelle se trouve votre espace de travail Amazon Managed Service for Prometheus.

   Par exemple, effectuez la modification suivante :

   ```
   AMP:
       remote_write_url: "https://aps-workspaces.us-east-2.amazonaws.com/workspaces/
   {workspaceId}/api/v1/remote_write"
       region: us-east-2
   ```

   Modifiez les noms du flux de diffusion Firehose et du compartiment Amazon S3 à votre guise.
2. Pour créer le code Lambda AWS CDK et le code Lambda, exécutez la recommandation suivante à la racine du dépôt :

   ```
   npm run build
   ```

   Cette étape de génération garantit que le binaire Go Lambda est créé et déploie le CDK sur. CloudFormation
3. Pour terminer le déploiement, passez en revue et acceptez les modifications IAM requises par la pile.
4. (Facultatif) Vous pouvez vérifier que la pile a été créée en exécutant la commande suivante.

   ```
   aws cloudformation list-stacks
   ```

Une pile nommée `CDK Stack` figurera dans la liste.

Création d'un CloudWatch stream Amazon

Maintenant que vous disposez d'une fonction lambda pour gérer les métriques, vous pouvez créer le flux de métriques depuis Amazon CloudWatch.

Pour créer un flux CloudWatch de statistiques

1. Accédez à la CloudWatch console, à https://console.aws.amazon.com/cloudwatch/ l'adresse #metric -streams:StreamsList, puis sélectionnez Créer un flux métrique.
2. Sélectionnez les métriques requises, toutes les métriques ou uniquement celles des espaces de noms sélectionnés.
3. Sous `Configuration`, choisissez Sélectionner un Firehose existant appartenant à votre compte.
4. Vous utiliserez le flux Firehose créé précédemment par CDK. Dans le menu déroulant Sélectionner votre flux Kinesis Data Firehose, sélectionnez le flux créé précédemment. Il aura un nom tel que `CdkStack-KinesisFirehoseStream123456AB-sample1234`.
5. Modifiez le format de sortie en JSON.
6. Donnez au flux de métriques un nom significatif.
7. Choisissez Créer un flux de métriques.
8. (Facultatif) Pour vérifier l'invocation de la fonction Lambda, accédez à la console Lambda et choisissez la fonction `KinesisMessageHandler`. Sélectionnez l'onglet Surveiller et le sous-onglet Journaux. Des entrées de la fonction Lambda devraient être déclenchées dans Invocations récentes.

> **Note**
>
> Il peut s'écouler jusqu'à 5 minutes avant que les invocations ne commencent à s'afficher dans l'onglet Surveiller.

Vos statistiques sont désormais diffusées d'Amazon CloudWatch vers Amazon Managed Service for Prometheus.

Nettoyage

Vous pouvez nettoyer les ressources qui ont été utilisées dans cet exemple. La procédure suivante explique comment procéder. Elle permettra d'arrêter le flux de métriques que vous avez créé.

Pour nettoyer des ressources

1. Commencez par supprimer la CloudFormation pile à l'aide des commandes suivantes :

```
cd cdk
cdk destroy
```

2. Supprimer l'espace de travail Amazon Managed Service for Prometheus :

```
aws amp delete-workspace --workspace-id \
    `aws amp list-workspaces --alias prometheus-sample-app --query
 'workspaces[0].workspaceId' --output text`
```

3. Enfin, supprimez le flux CloudWatch métrique Amazon à l'aide de la CloudWatch console Amazon.

Sécurité dans Amazon Managed Service for Prometheus

La sécurité du cloud AWS est la priorité absolue. En tant que AWS client, vous bénéficiez de centres de données et d'architectures réseau conçus pour répondre aux exigences des entreprises les plus sensibles en matière de sécurité.

La sécurité est une responsabilité partagée entre vous AWS et vous. Le modèle de responsabilité partagée décrit ceci comme la sécurité du cloud et la sécurité dans le cloud :

- Sécurité du cloud : AWS est chargée de protéger l'infrastructure qui exécute les AWS services dans le AWS cloud. AWS vous fournit également des services que vous pouvez utiliser en toute sécurité. Des auditeurs tiers testent et vérifient régulièrement l'efficacité de notre sécurité dans le cadre des programmes de AWS conformité Programmes de de conformité. Pour en savoir plus sur les programmes de conformité qui s'appliquent à Amazon Managed Service for Prometheus, AWS consultez la section Services concernés par le programme de conformité Services concernés par AWS
- Sécurité dans le cloud — Votre responsabilité est déterminée par le AWS service que vous utilisez. Vous êtes également responsable d'autres facteurs, y compris de la sensibilité de vos données, des exigences de votre entreprise, ainsi que de la législation et de la réglementation applicables.

Cette documentation vous aide à comprendre comment appliquer le modèle de responsabilité partagée lors de l'utilisation d'Amazon Managed Service for Prometheus. Les rubriques suivantes vous montrent comment configurer Amazon Managed Service for Prometheus pour répondre à vos objectifs de sécurité et de conformité. Vous apprendrez également à utiliser d'autres AWS services qui vous aident à surveiller et à sécuriser vos ressources Amazon Managed Service for Prometheus.

Rubriques

- Protection des données dans Amazon Managed Service for Prometheus
- Gestion de l'identité et des accès dans Amazon Managed Service for Prometheus
- Autorisations et politiques IAM
- Validation de la conformité pour Amazon Managed Service for Prometheus
- Résilience dans Amazon Managed Service for Prometheus
- Sécurité de l'infrastructure dans Amazon Managed Service for Prometheus
- Utilisation de rôles liés à un service pour Amazon Managed Service for Prometheus

- Journalisation des appels d'API Amazon Managed Service pour Prometheus à l'aide de AWS CloudTrail
- Configuration de rôles IAM pour les comptes de service
- Utilisation d'Amazon Managed Service for Prometheus avec des points de terminaison de VPC d'interface

Protection des données dans Amazon Managed Service for Prometheus

Le modèle de responsabilité AWS partagée (modèle de) s'applique à la protection des données dans Amazon Managed Service for Prometheus. Comme décrit dans ce modèle, AWS est chargé de protéger l'infrastructure mondiale qui gère tous les AWS Cloud. La gestion du contrôle de votre contenu hébergé sur cette infrastructure relève de votre responsabilité. Vous êtes également responsable des tâches de configuration et de gestion de la sécurité des Services AWS que vous utilisez. Pour plus d'informations sur la confidentialité des données, consultez la FAQ sur la confidentialité des données et les . Pour plus d'informations sur la protection des données en Europe, consultez le Centre du règlement général sur la protection des données (RGPD).

À des fins de protection des données, nous vous recommandons de protéger les Compte AWS informations d'identification et de configurer les utilisateurs individuels avec AWS IAM Identity Center ou Gestion des identités et des accès AWS (IAM). Ainsi, chaque utilisateur se voit attribuer uniquement les autorisations nécessaires pour exécuter ses tâches. Nous vous recommandons également de sécuriser vos données comme indiqué ci-dessous :

- Utilisez l'authentification multifactorielle (MFA) avec chaque compte.
- SSL/TLS À utiliser pour communiquer avec AWS les ressources. Nous exigeons TLS 1.2 et recommandons TLS 1.3.
- Configurez l'API et la journalisation de l'activité des utilisateurs avec AWS CloudTrail. Pour plus d'informations sur l'utilisation des CloudTrail sentiers pour capturer AWS des activités, consultez la section Utilisation des CloudTrail sentiers dans le guide de AWS CloudTrail l'utilisateur.
- Utilisez des solutions de AWS chiffrement, ainsi que tous les contrôles de sécurité par défaut qu'ils contiennent Services AWS.
- Utilisez des services de sécurité gérés avancés tels qu'Amazon Macie, qui contribuent à la découverte et à la sécurisation des données sensibles stockées dans Amazon S3.

- Si vous avez besoin de modules cryptographiques validés par la norme FIPS 140-3 pour accéder AWS via une interface de ligne de commande ou une API, utilisez un point de terminaison FIPS. Pour plus d'informations sur les points de terminaison FIPS disponibles, consultez Norme FIPS (Federal Information Processing Standard) 140-3.

Nous vous recommandons fortement de ne jamais placer d'informations confidentielles ou sensibles, telles que les adresses e-mail de vos clients, dans des balises ou des champs de texte libre tels que le champ Nom. Cela inclut lorsque vous travaillez avec Amazon Managed Service pour Prometheus ou avec un Services AWS autre service à l'aide de la console, de l'API AWS CLI ou des SDK. AWS Toutes les données que vous entrez dans des balises ou des champs de texte de forme libre utilisés pour les noms peuvent être utilisées à des fins de facturation ou dans les journaux de diagnostic. Si vous fournissez une adresse URL à un serveur externe, nous vous recommandons fortement de ne pas inclure d'informations d'identification dans l'adresse URL permettant de valider votre demande adressée à ce serveur.

Rubriques

- Données collectées par Amazon Managed Service for Prometheus
- Chiffrement au repos

Données collectées par Amazon Managed Service for Prometheus

Amazon Managed Service for Prometheus collecte et stocke les métriques opérationnelles que vous configurez pour être envoyées depuis les serveurs Prometheus exécutés sur votre compte vers Amazon Managed Service for Prometheus. Ces données comprennent les éléments suivants :

- Valeurs de métriques
- Étiquettes de métriques (ou paires clé-valeur arbitraires) qui aident à identifier et à classer les données
- Horodatages pour les échantillons de données

Les ID uniques des locataires permettent d'isoler les données des différents clients. Ces ID limitent les données clients accessibles. Les clients ne peuvent pas modifier les ID des locataires.

Amazon Managed Service for Prometheus chiffre les données qu'il stocke AWS Key Management Service à l'aide des clés ().AWS KMS Amazon Managed Service for Prometheus gère ces clés.

> **Note**
>
> Amazon Managed Service for Prometheus prend en charge la création de clés gérées par le client pour chiffrer vos données. Pour plus d'informations sur les clés qu'Amazon Managed Service for Prometheus utilise par défaut et sur la façon d'utiliser vos propres clés gérées par le client, consultez. Chiffrement au repos

Les données en transit sont automatiquement chiffrées à l'aide du protocole HTTPS. Amazon Managed Service for Prometheus sécurise les connexions entre les zones de disponibilité d'une région à l'aide du protocole HTTPS en AWS interne.

Chiffrement au repos

Par défaut, Amazon Managed Service for Prometheus vous fournit automatiquement le chiffrement au repos, en AWS utilisant les clés de chiffrement que vous possédez.

- AWS clés détenues : Amazon Managed Service for Prometheus utilise ces clés pour chiffrer automatiquement les données téléchargées sur votre espace de travail. Vous ne pouvez pas consulter, gérer ou utiliser AWS les clés que vous possédez, ni auditer leur utilisation. Toutefois, vous n'avez pas besoin de prendre de mesure ou de modifier les programmes pour protéger les clés qui chiffrent vos données. Pour plus d'informations, consultez Clés détenues par AWS dans le Guide du développeur AWS Key Management Service .

Le chiffrement au repos permet de réduire la charge opérationnelle et la complexité liées à la protection des données sensibles des clients, telles que les informations personnelles identifiables. Il vous permet de créer des applications sécurisées qui répondent aux exigences réglementaires et de conformité strictes en matière de chiffrement.

Vous pouvez également choisir d'utiliser une clé gérée par le client lorsque vous créez votre espace de travail :

- Clés gérées par le client : Amazon Managed Service for Prometheus prend en charge l'utilisation d'une clé symétrique gérée par le client, que vous créez, détenez et gérez pour chiffrer les données de l'espace de travail. Étant donné que vous avez le contrôle total du chiffrement, vous pouvez effectuer les tâches suivantes :
  - Établissement et gestion des stratégies de clé
  - Établissement et gestion des politiques IAM et des octrois

- Activation et désactivation des stratégies de clé
- Rotation des matériaux de chiffrement de clé
- Ajout de balises
- Création d'alias de clé
- Planification des clés pour la suppression

Pour plus d'informations, consultez Clés gérées par le client dans le Manuel du développeur AWS Key Management Service .

Choisissez d'utiliser avec soin les clés gérées par le client ou les clés AWS détenues par le client. Les espaces de travail créés avec des clés gérées par le client ne peuvent pas être convertis ultérieurement pour utiliser des clés AWS détenues (et vice versa).

> **Note**
>
> Amazon Managed Service for Prometheus active automatiquement le chiffrement au repos à AWS l'aide de clés détenues afin de protéger gratuitement vos données.
> Toutefois, AWS KMS des frais s'appliquent pour l'utilisation d'une clé gérée par le client. Pour plus d'informations sur la tarification, consultez Tarification d'AWS Key Management Service.

Pour plus d'informations AWS KMS, voir Qu'est-ce que c'est AWS Key Management Service ?

> **Note**
>
> Les espaces de travail créés avec les clés gérées par le client ne peuvent pas utiliser les collecteurs gérés AWS pour l'ingestion.

Comment Amazon Managed Service for Prometheus utilise les subventions dans AWS KMS

Amazon Managed Service for Prometheus requiert trois attributions pour utiliser votre clé gérée par le client.

Lorsque vous créez un espace de travail Amazon Managed Service for Prometheus chiffré à l'aide d'une clé gérée par le client, Amazon Managed Service for Prometheus crée les trois subventions

en votre nom en envoyant des demandes à. CreateGrant AWS KMS Les subventions AWS KMS sont utilisées pour permettre à Amazon Managed Service for Prometheus d'accéder à la clé KMS de votre compte, même si vous n'êtes pas appelé directement en votre nom (par exemple, lorsque vous stockez des données de métriques extraites d'un cluster Amazon EKS).

Amazon Managed Service for Prometheus nécessite que les attributions utilisent la clé gérée par le client pour les opérations internes suivantes :

- Envoyez DescribeKeydes demandes AWS KMS à pour vérifier que la clé KMS symétrique gérée par le client fournie lors de la création d'un espace de travail est valide.
- Envoyez GenerateDataKeydes demandes AWS KMS à pour générer des clés de données chiffrées par votre clé gérée par le client.
- Envoyez des demandes de déchiffrement AWS KMS à pour déchiffrer les clés de données chiffrées afin qu'elles puissent être utilisées pour chiffrer vos données.

Amazon Managed Service for Prometheus octroie trois autorisations à AWS KMS la clé, ce qui permet à Amazon Managed Service for Prometheus d'utiliser la clé en votre nom. Vous pouvez supprimer l'accès à la clé en modifiant la politique relative aux clés, en désactivant la clé ou en révoquant l'attribution. Vous devez comprendre les conséquences de ces actions avant de les exécuter. Cela peut entraîner une perte de données dans votre espace de travail.

Si vous supprimez de quelque façon que ce soit l'accès à l'une des attributions, Amazon Managed Service for Prometheus ne pourra accéder à aucune des données chiffrées par la clé gérée par le client, ni stocker les nouvelles données envoyées à l'espace de travail, ce qui affecte les opérations dépendant de ces données. Les nouvelles données envoyées à l'espace de travail ne seront pas accessibles et risquent d'être définitivement perdues.

⚠ Warning

- Si vous désactivez la clé ou si vous supprimez l'accès à Amazon Managed Service for Prometheus dans la politique relative aux clés, les données de l'espace de travail ne sont plus accessibles. Les nouvelles données envoyées à l'espace de travail ne seront pas accessibles et risquent d'être définitivement perdues.

  Vous pouvez accéder aux données de l'espace de travail et recommencer à recevoir de nouvelles données en rétablissant l'accès d'Amazon Managed Service for Prometheus à la clé.

- Si vous révoquez une attribution, elle ne peut pas être recréée et les données de l'espace de travail sont définitivement perdues.

Étape 1 : Créer une clé gérée par le client

Vous pouvez créer une clé symétrique gérée par le client à l'aide de Console de gestion AWS, ou des AWS KMS API. Il n'est pas nécessaire que la clé se trouve sur le même compte que l'espace de travail Amazon Managed Service for Prometheus, sous réserve que vous fournissiez l'accès correct via la politique, comme décrit ci-dessous.

Pour créer une clé symétrique gérée par le client

Suivez les étapes de la rubrique Création d'une clé symétrique gérée par le client dans le Guide du développeur AWS Key Management Service .

Stratégie de clé

Les stratégies de clé contrôlent l'accès à votre clé gérée par le client. Chaque clé gérée par le client doit avoir exactement une stratégie de clé, qui contient des instructions qui déterminent les personnes pouvant utiliser la clé et comment elles peuvent l'utiliser. Lorsque vous créez votre clé gérée par le client, vous pouvez spécifier une stratégie de clé. Pour plus d'informations, consultez Gestion de l'accès aux clés gérées par le client dans le Guide du développeur AWS Key Management Service .

Pour utiliser votre clé gérée par le client avec les espaces de travail Amazon Managed Service for Prometheus, les opérations d'API suivantes doivent être autorisées dans la politique des clés :

- `kms:CreateGrant` : ajoute une attribution à une clé gérée par le client. Octroie un accès de contrôle à une clé KMS spécifiée, ce qui permet d'accéder aux opérations d'attribution requises par Amazon Managed Service for Prometheus. Pour plus d'informations, consultez Utilisation des attributions dans le Guide du développeur AWS Key Management Service .

  Amazon Managed Service for Prometheus peut ainsi exécuter les tâches suivantes :

  - Appelez `GenerateDataKey` pour générer une clé de données chiffrée et la stocker, car la clé de données n'est pas immédiatement utilisée pour chiffrer.
  - Appelez `Decrypt` pour utiliser la clé de données chiffrée stockée afin d'accéder aux données chiffrées.

- `kms:DescribeKey` : fournit les détails des clés gérées par le client pour permettre à Amazon Managed Service for Prometheus de valider la clé.

Voici quelques exemples de déclarations de politique pour Amazon Managed Service for Prometheus :

```
 "Statement" : [
   {
     "Sid" : "Allow access to Amazon Managed Service for Prometheus principal within
your account",
     "Effect" : "Allow",
     "Principal" : {
       "AWS" : "*"
     },
     "Action" : [
       "kms:DescribeKey",
       "kms:CreateGrant",
       "kms:GenerateDataKey",
       "kms:Decrypt"
     ],
     "Resource" : "*",
     "Condition" : {
       "StringEquals" : {
         "kms:ViaService" : "aps.region.amazonaws.com",
         "kms:CallerAccount" : "111122223333"
       }
   },
   {
     "Sid": "Allow access for key administrators - not required for Amazon Managed
Service for Prometheus",
     "Effect": "Allow",
     "Principal": {
       "AWS": "arn:aws:iam::111122223333:root"
      },
     "Action" : [
       "kms:*"
      ],
     "Resource": "arn:aws:kms:region:111122223333:key/key_ID"
   },
   <other statements needed for other non-Amazon Managed Service for Prometheus
scenarios>
 ]
```

- Pour plus d'informations sur la spécification d'autorisations dans une politique, consultez le Guide du développeur AWS Key Management Service .
- Pour plus d'informations sur le dépannage des clés d'accès, consultez le Guide du développeur AWS Key Management Service .

Étape 2 : Spécification d'une clé gérée par le client pour Amazon Managed Service for Prometheus

Lorsque vous créez un espace de travail, vous pouvez spécifier la clé gérée par le client en saisissant un ARN de clé KMS, qu'Amazon Managed Service for Prometheus utilise pour chiffrer les données stockées dans l'espace de travail.

Étape 3 : Accès aux données depuis d'autres services, tels qu'Amazon Managed Grafana

Cette étape est facultative. Elle n'est requise que si vous devez accéder à vos données Amazon Managed Service for Prometheus depuis un autre service.

Vos données cryptées ne sont pas accessibles depuis d'autres services, à moins qu'ils n'aient également accès à la AWS KMS clé. Par exemple, si vous souhaitez utiliser Amazon Managed Grafana pour créer un tableau de bord ou une alerte concernant vos données, vous devez autoriser Amazon Managed Grafana à accéder à la clé.

Pour donner à Amazon Managed Grafana l'accès à votre clé gérée par le client

1. Dans votre liste d'espaces de travail Amazon Managed Grafana, sélectionnez le nom de l'espace de travail auquel vous souhaitez avoir accès à Amazon Managed Service for Prometheus. Cela vous montre des informations récapitulatives sur votre espace de travail Amazon Managed Grafana.
2. Notez le nom du rôle IAM utilisé par votre espace de travail. Le nom est au format`AmazonGrafanaServiceRole-<unique-id>`. La console affiche l'ARN complet du rôle. Vous spécifierez ce nom dans la AWS KMS console lors d'une étape ultérieure.
3. Dans votre liste de clés gérées par le AWS KMS client, choisissez la clé gérée par le client que vous avez utilisée lors de la création de votre espace de travail Amazon Managed Service for Prometheus. Cela ouvre la page des principaux détails de configuration.
4. À côté de Utilisateurs clés, sélectionnez le bouton Ajouter.

5. Dans la liste des noms, choisissez le rôle Amazon Managed Grafana IAM indiqué ci-dessus. Pour faciliter la recherche, vous pouvez également effectuer une recherche par nom.
6. Choisissez Ajouter pour ajouter le rôle IAM à la liste des utilisateurs clés.

Votre espace de travail Amazon Managed Grafana peut désormais accéder aux données de votre espace de travail Amazon Managed Service for Prometheus. Vous pouvez ajouter d'autres utilisateurs ou rôles aux utilisateurs principaux pour permettre à d'autres services d'accéder à votre espace de travail.

Contexte de chiffrement Amazon Managed Service for Prometheus

Un contexte de chiffrement est un ensemble facultatif de paires clé-valeur qui contient des informations contextuelles supplémentaires sur les données.

AWS KMS utilise le contexte de chiffrement comme données authentifiées supplémentaires pour prendre en charge le chiffrement authentifié. Lorsque vous incluez un contexte de chiffrement dans une demande de chiffrement de données, AWS KMS lie le contexte de chiffrement aux données chiffrées. Pour déchiffrer les données, vous devez inclure le même contexte de chiffrement dans la demande.

Contexte de chiffrement Amazon Managed Service for Prometheus

Amazon Managed Service for Prometheus utilise le même contexte de chiffrement dans AWS KMS toutes les opérations cryptographiques, où la clé et la valeur `aws:amp:arn` sont le nom de ressource Amazon (ARN) de l'espace de travail.

Example

```
"encryptionContext": {
    "aws:aps:arn": "arn:aws:aps:us-west-2:111122223333:workspace/ws-sample-1234-
abcd-56ef-7890abcd12ef"
}
```

Utilisation du contexte de chiffrement pour la surveillance

Lorsque vous utilisez une clé symétrique gérée par le client pour chiffrer les données de votre espace de travail, vous pouvez également utiliser le contexte de chiffrement dans les enregistrements et les journaux d'audit pour identifier la manière dont la clé gérée par le client est utilisée. Le contexte

de chiffrement apparaît également dans les journaux générés par AWS CloudTrail ou Amazon CloudWatch Logs.

Utilisation du contexte de chiffrement pour contrôler l'accès à votre clé gérée par le client

Vous pouvez utiliser le contexte de chiffrement dans les stratégies de clé et les politiques IAM comme `conditions` pour contrôler l'accès à votre clé symétrique gérée par le client. Vous pouvez également utiliser des contraintes de contexte de chiffrement dans un octroi.

Amazon Managed Service for Prometheus utilise une contrainte de contexte de chiffrement dans les octrois pour contrôler l'accès à la clé gérée par le client dans votre compte ou région. La contrainte d'octroi exige que les opérations autorisées par l'octroi utilisent le contexte de chiffrement spécifié.

Example

Vous trouverez ci-dessous des exemples de déclarations de stratégie de clé permettant d'accorder l'accès à une clé gérée par le client dans un contexte de chiffrement spécifique. La condition énoncée dans cette instruction de stratégie exige que les octrois comportent une contrainte de contexte de chiffrement qui spécifie le contexte de chiffrement.

```
{
    "Sid": "Enable DescribeKey",
    "Effect": "Allow",
    "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/ExampleReadOnlyRole"
     },
     "Action": "kms:DescribeKey",
     "Resource": "*"
},
{
     "Sid": "Enable CreateGrant",
     "Effect": "Allow",
     "Principal": {
         "AWS": "arn:aws:iam::111122223333:role/ExampleReadOnlyRole"
     },
     "Action": "kms:CreateGrant",
     "Resource": "*",
     "Condition": {
         "StringEquals": {
             "kms:EncryptionContext:aws:aps:arn": "arn:aws:aps:us-
west-2:111122223333:workspace/ws-sample-1234-abcd-56ef-7890abcd12ef"
          }
     }
```

```
}
```

Surveillance des clés de chiffrement pour Amazon Managed Service for Prometheus

Lorsque vous utilisez une clé gérée par le AWS KMS client avec vos espaces de travail Amazon Managed Service for Prometheus, vous pouvez utiliser AWS CloudTrailAmazon Logs pour suivre les demandes CloudWatch auxquelles Amazon Managed Service for Prometheus envoie. AWS KMS

Les exemples suivants sont AWS CloudTrail des événements pour`CreateGrant`, `GenerateDataKeyDecrypt`, et `DescribeKey` pour surveiller les opérations KMS appelées par Amazon Managed Service pour Prometheus afin d'accéder aux données chiffrées par votre clé gérée par le client :

CreateGrant

Lorsque vous utilisez une clé gérée par le AWS KMS client pour chiffrer votre espace de travail, Amazon Managed Service for Prometheus envoie `CreateGrant` trois demandes en votre nom pour accéder à la clé KMS que vous avez spécifiée. Les attributions créées par Amazon Managed Service for Prometheus sont spécifiques à la ressource associée à la clé AWS KMS gérée par le client.

L'exemple d'événement suivant enregistre une opération `CreateGrant` :

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "TESTANDEXAMPLE:Sampleuser01",
        "arn": "arn:aws:sts::111122223333:assumed-role/Admin/Sampleuser01",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE-KEY-ID1",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "TESTANDEXAMPLE:Sampleuser01",
                "arn": "arn:aws:sts::111122223333:assumed-role/Admin/Sampleuser01",
                "accountId": "111122223333",
                "userName": "Admin"
            },
            "webIdFederationData": {},
            "attributes": {
                "mfaAuthenticated": "false",
```

```
                "creationDate": "2021-04-22T17:02:00Z"
            }
        },
        "invokedBy": "aps.amazonaws.com"
    },
    "eventTime": "2021-04-22T17:07:02Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "CreateGrant",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "172.12.34.56",
    "userAgent": "ExampleDesktop/1.0 (V1; OS)",
    "requestParameters": {
        "retiringPrincipal": "aps.region.amazonaws.com",
        "operations": [
            "GenerateDataKey",
            "Decrypt",
            "DescribeKey"
        ],
        "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE",
        "granteePrincipal": "aps.region.amazonaws.com"
    },
    "responseElements": {
        "grantId":
 "0ab0ac0d0b000f00ea00cc0a0e00fc00bce000c000f0000000c0bc0a0000aaafSAMPLE"
    },
    "requestID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "eventID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "readOnly": false,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "eventCategory": "Management",
    "recipientAccountId": "111122223333"
}
```

GenerateDataKey

Lorsque vous activez une clé gérée par le AWS KMS client pour votre espace de travail, Amazon Managed Service for Prometheus crée une clé unique. Il envoie une `GenerateDataKey` demande AWS KMS qui spécifie la clé gérée par le AWS KMS client pour la ressource.

L'exemple d'événement suivant enregistre l'opération `GenerateDataKey` :


```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AWSService",
        "invokedBy": "aps.amazonaws.com"
    },
    "eventTime": "2021-04-22T17:07:02Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "GenerateDataKey",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "172.12.34.56",
    "userAgent": "ExampleDesktop/1.0 (V1; OS)",
    "requestParameters": {
        "encryptionContext": {
            "aws:aps:arn": "arn:aws:aps:us-west-2:111122223333:workspace/ws-
sample-1234-abcd-56ef-7890abcd12ef"
        },
        "keySpec": "AES_256",
        "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
    },
    "responseElements": null,
    "requestID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "eventID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
```

```
    "eventCategory": "Management",
    "recipientAccountId": "111122223333",
    "sharedEventID": "57f5dbee-16da-413e-979f-2c4c6663475e"
}
```

Decrypt

Lorsqu'une requête est générée sur un espace de travail chiffré, Amazon Managed Service for Prometheus appelle l'opération `Decrypt` pour utiliser la clé de données chiffrée stockée afin d'accéder aux données chiffrées.

L'exemple d'événement suivant enregistre l'opération `Decrypt` :

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AWSService",
        "invokedBy": "aps.amazonaws.com"
    },
    "eventTime": "2021-04-22T17:10:51Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "Decrypt",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "172.12.34.56",
    "userAgent": "ExampleDesktop/1.0 (V1; OS)",
    "requestParameters": {
        "encryptionContext": {
            "aws:aps:arn": "arn:aws:aps:us-west-2:111122223333:workspace/ws-
sample-1234-abcd-56ef-7890abcd12ef"
        },
        "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE",
        "encryptionAlgorithm": "SYMMETRIC_DEFAULT"
    },
    "responseElements": null,
    "requestID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "eventID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
```

```
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "eventCategory": "Management",
    "recipientAccountId": "111122223333",
    "sharedEventID": "dc129381-1d94-49bd-b522-f56a3482d088"
}
```

DescribeKey

Amazon Managed Service for Prometheus utilise l'opération `DescribeKey` pour vérifier si la clé AWS KMS gérée par le client associée à votre espace de travail existe dans le compte et dans la région.

L'exemple d'événement suivant enregistre l'opération `DescribeKey` :

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "TESTANDEXAMPLE:Sampleuser01",
        "arn": "arn:aws:sts::111122223333:assumed-role/Admin/Sampleuser01",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE-KEY-ID1",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "TESTANDEXAMPLE:Sampleuser01",
                "arn": "arn:aws:sts::111122223333:assumed-role/Admin/Sampleuser01",
                "accountId": "111122223333",
                "userName": "Admin"
            },
            "webIdFederationData": {},
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2021-04-22T17:02:00Z"
            }
        },
        "invokedBy": "aps.amazonaws.com"
    },
```

```
    "eventTime": "2021-04-22T17:07:02Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "DescribeKey",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "172.12.34.56",
    "userAgent": "ExampleDesktop/1.0 (V1; OS)",
    "requestParameters": {
        "keyId": "00dd0db0-0000-0000-ac00-b0c000SAMPLE"
    },
    "responseElements": null,
    "requestID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "eventID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "eventCategory": "Management",
    "recipientAccountId": "111122223333"
}
```


En savoir plus

Les ressources suivantes fournissent plus d'informations sur le chiffrement des données au repos.

- Pour plus d'informations sur les concepts de base AWS Key Management Service, consultez le Guide du développeur AWS Key Management Service .
- Pour plus d'informations sur les meilleures pratiques de sécurité pour AWS Key Management Service, consultez le guide du AWS Key Management Service développeur.

Gestion de l'identité et des accès dans Amazon Managed Service for Prometheus

Gestion des identités et des accès AWS (IAM) est un outil Service AWS qui permet à un administrateur de contrôler en toute sécurité l'accès aux AWS ressources. Des administrateurs IAM contrôlent les personnes qui peuvent être authentifiées (connectées) et autorisées (disposant d'autorisations) à utiliser des ressources Amazon Managed Service for Prometheus. IAM est un Service AWS outil que vous pouvez utiliser sans frais supplémentaires.

Rubriques

- Public ciblé
- Authentification par des identités
- Gestion de l'accès à l'aide de politiques
- Utilisation d'Amazon Managed Service for Prometheus avec IAM
- Exemples de politiques basées sur l'identité pour Amazon Managed Service for Prometheus
- Résolution des problèmes liés à l'identité et aux accès dans Amazon Managed Service for Prometheus

Public ciblé

La façon dont vous utilisez Gestion des identités et des accès AWS (IAM) varie en fonction de votre rôle :

- Utilisateur du service : demandez des autorisations à votre administrateur si vous ne pouvez pas accéder aux fonctionnalités (voir Résolution des problèmes liés à l'identité et aux accès dans Amazon Managed Service for Prometheus)
- Administrateur du service : déterminez l'accès des utilisateurs et soumettez les demandes d'autorisation (voir Utilisation d'Amazon Managed Service for Prometheus avec IAM)
- Administrateur IAM : rédigez des politiques pour gérer l'accès (voir Exemples de politiques basées sur l'identité pour Amazon Managed Service for Prometheus)

Authentification par des identités

L'authentification est la façon dont vous vous connectez à AWS l'aide de vos informations d'identification. Vous devez être authentifié en tant qu'utilisateur IAM ou en assumant un rôle IAM. Utilisateur racine d'un compte AWS

Vous pouvez vous connecter en tant qu'identité fédérée à l'aide d'informations d'identification provenant d'une source d'identité telle que AWS IAM Identity Center (IAM Identity Center), d'une

authentification unique ou d'informations d'identification. Google/Facebook Pour plus d'informations sur la connexion, consultez Connexion à votre Compte AWS dans le Guide de l'utilisateur Connexion à AWS .

Pour l'accès par programmation, AWS fournit un SDK et une CLI pour signer les demandes de manière cryptographique. Pour plus d'informations, consultez Signature AWS Version 4 pour les demandes d'API dans le Guide de l'utilisateur IAM.

Compte AWS utilisateur root

Lorsque vous créez un Compte AWS, vous commencez par une seule identité de connexion appelée utilisateur Compte AWS root qui dispose d'un accès complet à toutes Services AWS les ressources. Il est vivement déconseillé d'utiliser l'utilisateur racine pour vos tâches quotidiennes. Pour les tâches qui requièrent des informations d'identification de l'utilisateur racine, consultez Tâches qui requièrent les informations d'identification de l'utilisateur racine dans le Guide de l'utilisateur IAM.

Identité fédérée

Il est recommandé d'obliger les utilisateurs humains à utiliser la fédération avec un fournisseur d'identité pour accéder à Services AWS l'aide d'informations d'identification temporaires.

Une identité fédérée est un utilisateur provenant de l'annuaire de votre entreprise, de votre fournisseur d'identité Web ou Directory Service qui y accède à Services AWS l'aide d'informations d'identification provenant d'une source d'identité. Les identités fédérées assument des rôles qui fournissent des informations d'identification temporaires.

Pour une gestion des accès centralisée, nous vous recommandons d'utiliser AWS IAM Identity Center. Pour plus d'informations, consultez Qu'est-ce que IAM Identity Center ? dans le Guide de l'utilisateur AWS IAM Identity Center .

Utilisateurs et groupes IAM

Un utilisateur IAM est une identité qui dispose d'autorisations spécifiques pour une seule personne ou application. Nous vous recommandons d'utiliser ces informations d'identification temporaires au lieu des utilisateurs IAM avec des informations d'identification à long terme. Pour plus d'informations, voir Exiger des utilisateurs humains qu'ils utilisent la fédération avec un fournisseur d'identité pour accéder à AWS l'aide d'informations d'identification temporaires dans le guide de l'utilisateur IAM.

Les groupes IAM spécifient une collection d'utilisateurs IAM et permettent de gérer plus facilement les autorisations pour de grands ensembles d'utilisateurs. Pour plus d'informations, consultez Cas d'utilisation pour les utilisateurs IAM dans le Guide de l'utilisateur IAM.

Rôles IAM

Un rôle IAM est une identité dotée d'autorisations spécifiques qui fournit des informations d'identification temporaires. Vous pouvez assumer un rôle en passant d'un rôle d'utilisateur à un rôle IAM (console) ou en appelant une opération d' AWS API AWS CLI ou d'API. Pour plus d'informations, consultez Méthodes pour endosser un rôle dans le Guide de l'utilisateur IAM.

Les rôles IAM sont utiles pour l'accès des utilisateurs fédérés, les autorisations temporaires des utilisateurs IAM, les accès intercompte, les accès entre services et les applications exécutées sur Amazon EC2. Pour plus d'informations, consultez Accès intercompte aux ressources dans IAM dans le Guide de l'utilisateur IAM.

Gestion de l'accès à l'aide de politiques

Vous contrôlez l'accès en AWS créant des politiques et en les associant à AWS des identités ou à des ressources. Une politique définit les autorisations lorsqu'elles sont associées à une identité ou à une ressource. AWS évalue ces politiques lorsqu'un directeur fait une demande. La plupart des politiques sont stockées AWS sous forme de documents JSON. Pour plus d'informations les documents de politique JSON, consultez Vue d'ensemble des politiques JSON dans le Guide de l'utilisateur IAM.

À l'aide de politiques, les administrateurs précisent qui a accès à quoi en définissant quel principal peut effectuer des actions sur quelles ressources et dans quelles conditions.

Par défaut, les utilisateurs et les rôles ne disposent d'aucune autorisation. Un administrateur IAM crée des politiques IAM et les ajoute aux rôles, que les utilisateurs peuvent ensuite assumer. Les politiques IAM définissent les autorisations quelle que soit la méthode que vous utilisez pour exécuter l'opération.

Politiques basées sur l'identité

Les stratégies basées sur l'identité sont des documents de stratégie d'autorisations JSON que vous attachez à une identité (utilisateur, groupe ou rôle). Ces politiques contrôlent les actions que peuvent exécuter ces identités, sur quelles ressources et dans quelles conditions. Pour découvrir comment créer une politique basée sur l'identité, consultez Définition d'autorisations IAM personnalisées avec des politiques gérées par le client dans le Guide de l'utilisateur IAM.

Les politiques basées sur l'identité peuvent être des politiques intégrées (intégrées directement dans une seule identité) ou des politiques gérées (politiques autonomes associées à plusieurs identités).

Pour découvrir comment choisir entre des politiques gérées et en ligne, consultez Choix entre les politiques gérées et les politiques en ligne dans le Guide de l'utilisateur IAM.

Politiques basées sur les ressources

Les politiques basées sur les ressources sont des documents de politique JSON que vous attachez à une ressource. Les exemples incluent les politiques de confiance de rôle IAM et les stratégies de compartiment Amazon S3. Dans les services qui sont compatibles avec les politiques basées sur les ressources, les administrateurs de service peuvent les utiliser pour contrôler l'accès à une ressource spécifique. Vous devez spécifier un principal dans une politique basée sur les ressources.

Les politiques basées sur les ressources sont des politiques en ligne situées dans ce service. Vous ne pouvez pas utiliser les politiques AWS gérées par IAM dans une stratégie basée sur les ressources.

Autres types de politique

AWS prend en charge des types de politiques supplémentaires qui peuvent définir les autorisations maximales accordées par les types de politiques les plus courants :

- Limites d'autorisations : une limite des autorisations définit le nombre maximum d'autorisations qu'une politique basée sur l'identité peut accorder à une entité IAM. Pour plus d'informations, consultez Limites d'autorisations pour des entités IAM dans le Guide de l'utilisateur IAM.
- Politiques de contrôle des services (SCPs) — Spécifiez les autorisations maximales pour une organisation ou une unité organisationnelle dans AWS Organizations. Pour plus d'informations, consultez Politiques de contrôle de service dans le Guide de l'utilisateur AWS Organizations .
- Politiques de contrôle des ressources (RCPs) : définissez le maximum d'autorisations disponibles pour les ressources de vos comptes. Pour plus d'informations, voir Politiques de contrôle des ressources (RCPs) dans le guide de AWS Organizations l'utilisateur.
- Politiques de session : politiques avancées que vous passez en tant que paramètre lorsque vous créez par programmation une session temporaire pour un rôle ou un utilisateur fédéré. Pour plus d'informations, consultez Politiques de session dans le Guide de l'utilisateur IAM.

Plusieurs types de politique

Lorsque plusieurs types de politiques s'appliquent à la requête, les autorisations en résultant sont plus compliquées à comprendre. Pour savoir comment AWS déterminer s'il faut autoriser

une demande lorsque plusieurs types de politiques sont impliqués, consultez la section Logique d'évaluation des politiques dans le guide de l'utilisateur IAM.

Utilisation d'Amazon Managed Service for Prometheus avec IAM

Avant d'utiliser IAM pour gérer l'accès à Amazon Managed Service for Prometheus, découvrez les fonctionnalités IAM qui peuvent être utilisées avec Amazon Managed Service for Prometheus.

Fonctionnalités IAM pouvant être utilisées avec Amazon Managed Service for Prometheus

Fonctionnalité IAM	Prise en charge d'Amazon Managed Service for Prometheus
Politiques basées sur l'identité	Oui
Politiques basées sur les ressources	Oui
Actions de politique	Oui
Ressources de politique	Oui
Clés de condition d'une politique	Non
ACLs	Non
ABAC (étiquettes dans les politiques)	Oui
Informations d'identification temporaires	Oui
Transfert des sessions d'accès (FAS)	Non
Rôles du service	Non
Rôles liés à un service	Oui

Pour obtenir une vue d'ensemble de la façon dont Amazon Managed Service for Prometheus et les AWS autres services fonctionnent avec la plupart des fonctionnalités IAM, AWS consultez les services compatibles avec IAM dans le guide de l'utilisateur IAM.

Politiques basées sur l'identité pour Amazon Managed Service for Prometheus

Prend en charge les politiques basées sur l'identité : oui

Les politiques basées sur l'identité sont des documents de politique d'autorisations JSON que vous pouvez attacher à une identité telle qu'un utilisateur, un groupe d'utilisateurs ou un rôle IAM. Ces politiques contrôlent quel type d'actions des utilisateurs et des rôles peuvent exécuter, sur quelles ressources et dans quelles conditions. Pour découvrir comment créer une politique basée sur l'identité, consultez Définition d'autorisations IAM personnalisées avec des politiques gérées par le client dans le Guide de l'utilisateur IAM.

Avec les politiques IAM basées sur l'identité, vous pouvez spécifier des actions et ressources autorisées ou refusées, ainsi que les conditions dans lesquelles les actions sont autorisées ou refusées. Pour découvrir tous les éléments que vous utilisez dans une politique JSON, consultez Références des éléments de politique JSON IAM dans le Guide de l'utilisateur IAM.

Exemples de politiques basées sur l'identité pour Amazon Managed Service for Prometheus

Pour afficher des exemples de politiques basées sur l'identité pour Amazon Managed Service for Prometheus, consultez la section Exemples de politiques basées sur l'identité pour Amazon Managed Service for Prometheus.

Politiques basées sur les ressources pour Amazon Managed Service for Prometheus

Prend en charge les politiques basées sur les ressources : oui

Les politiques basées sur les ressources sont des documents de politique JSON que vous attachez à une ressource. Par exemple, les politiques de confiance de rôle IAM et les politiques de compartiment Amazon S3 sont des politiques basées sur les ressources. Dans les services qui sont compatibles avec les politiques basées sur les ressources, les administrateurs de service peuvent les utiliser pour contrôler l'accès à une ressource spécifique. Pour la ressource dans laquelle se trouve la politique, cette dernière définit quel type d'actions un principal spécifié peut effectuer sur cette ressource et dans quelles conditions. Vous devez spécifier un principal dans une politique basée sur les ressources. Les principaux peuvent inclure des comptes, des utilisateurs, des rôles, des utilisateurs fédérés ou. Services AWS

Pour permettre un accès intercompte, vous pouvez spécifier un compte entier ou des entités IAM dans un autre compte en tant que principal dans une politique basée sur les ressources. Pour plus

d'informations, consultez Accès intercompte aux ressources dans IAM dans le Guide de l'utilisateur IAM.

Actions de politique pour Amazon Managed Service for Prometheus

Prend en charge les actions de politique : oui

Les administrateurs peuvent utiliser les politiques AWS JSON pour spécifier qui a accès à quoi. C'est-à-dire, quel principal peut effectuer des actions sur quelles ressources et dans quelles conditions.

L'élément `Action` d'une politique JSON décrit les actions que vous pouvez utiliser pour autoriser ou refuser l'accès à une politique. Intégration d'actions dans une politique afin d'accorder l'autorisation d'exécuter les opérations associées.

Pour afficher la liste des actions Amazon Managed Service for Prometheus, consultez la section Actions définies par Amazon Managed Service for Prometheus dans Référence de l'autorisation de service.

Les actions de politique dans Amazon Managed Service for Prometheus utilisent le préfixe suivant avant l'action :

```
aps
```

Pour indiquer plusieurs actions dans une seule déclaration, séparez-les par des virgules.

```
"Action": [
      "aps:action1",
      "aps:action2"
         ]
```

Pour afficher des exemples de politiques basées sur l'identité pour Amazon Managed Service for Prometheus, consultez la section Exemples de politiques basées sur l'identité pour Amazon Managed Service for Prometheus.

Ressources de politique pour Amazon Managed Service for Prometheus

Prend en charge les ressources de politique : oui

Les administrateurs peuvent utiliser les politiques AWS JSON pour spécifier qui a accès à quoi. C'est-à-dire, quel principal peut effectuer des actions sur quelles ressources et dans quelles conditions.

L'élément de politique JSON `Resource` indique le ou les objets auxquels l'action s'applique. Il est recommandé de définir une ressource à l'aide de son Amazon Resource Name (ARN). Pour les actions qui ne sont pas compatibles avec les autorisations de niveau ressource, utilisez un caractère générique (*) afin d'indiquer que l'instruction s'applique à toutes les ressources.

```
"Resource": "*"
```

Pour consulter la liste des types de ressources Amazon Managed Service for Prometheus et ARNs leurs caractéristiques, consultez la section Ressources définies par Amazon Managed Service for Prometheus dans le Service Authorization Reference. Pour savoir les actions avec lesquelles vous pouvez spécifier l'ARN de chaque ressource, consultez la section Actions définies par Amazon Managed Service for Prometheus.

Pour afficher des exemples de politiques basées sur l'identité pour Amazon Managed Service for Prometheus, consultez la section Exemples de politiques basées sur l'identité pour Amazon Managed Service for Prometheus.

Clés de condition d'une politique pour Amazon Managed Service for Prometheus

Prend en charge les clés de condition de politique spécifiques au service : non

Les administrateurs peuvent utiliser les politiques AWS JSON pour spécifier qui a accès à quoi. C'est-à-dire, quel principal peut effectuer des actions sur quelles ressources et dans quelles conditions.

L'élément `Condition` indique à quel moment les instructions s'exécutent en fonction de critères définis. Vous pouvez créer des expressions conditionnelles qui utilisent des opérateurs de condition, tels que les signes égal ou inférieur à, pour faire correspondre la condition de la politique aux valeurs de la demande. Pour voir toutes les clés de condition AWS globales, voir les clés de contexte de condition AWS globales dans le guide de l'utilisateur IAM.

Pour afficher la liste des clés de condition Amazon Managed Service for Prometheus, consultez la section Condition keys for Amazon Managed Service for Prometheus dans Référence de l'autorisation de service. Pour savoir avec quelles actions et ressources vous pouvez utiliser une clé de condition, consultez la section Actions définies par Amazon Managed Service for Prometheus.

Pour afficher des exemples de politiques basées sur l'identité pour Amazon Managed Service for Prometheus, consultez la section Exemples de politiques basées sur l'identité pour Amazon Managed Service for Prometheus.

Listes de contrôle d'accès (ACLs) dans Amazon Managed Service for Prometheus

Supports ACLs : Non

Les listes de contrôle d'accès (ACLs) contrôlent les principaux (membres du compte, utilisateurs ou rôles) autorisés à accéder à une ressource. ACLs sont similaires aux politiques basées sur les ressources, bien qu'elles n'utilisent pas le format de document de politique JSON.

Contrôle d'accès basé sur les attributs (ABAC) avec Amazon Managed Service for Prometheus

Prise en charge d'ABAC (balises dans les politiques) : Oui

Le contrôle d'accès par attributs (ABAC) est une stratégie d'autorisation qui définit les autorisations en fonction des attributs appelés balises. Vous pouvez associer des balises aux entités et aux AWS ressources IAM, puis concevoir des politiques ABAC pour autoriser les opérations lorsque la balise du principal correspond à la balise de la ressource.

Pour contrôler l'accès basé sur des étiquettes, vous devez fournir les informations d'étiquette dans l'élément de condition d'une politique utilisant les clés de condition `aws:ResourceTag/`*`key-name`*, `aws:RequestTag/`*`key-name`* ou `aws:TagKeys`.

Si un service prend en charge les trois clés de condition pour tous les types de ressources, alors la valeur pour ce service est Oui. Si un service prend en charge les trois clés de condition pour certains types de ressources uniquement, la valeur est Partielle.

Pour plus d'informations sur ABAC, consultez Définition d'autorisations avec l'autorisation ABAC dans le Guide de l'utilisateur IAM. Pour accéder à un didacticiel décrivant les étapes de configuration de l'ABAC, consultez Utilisation du contrôle d'accès par attributs (ABAC) dans le Guide de l'utilisateur IAM.

Utilisation d'informations d'identification temporaires avec Amazon Managed Service for Prometheus

Prend en charge les informations d'identification temporaires : oui

Les informations d'identification temporaires fournissent un accès à court terme aux AWS ressources et sont automatiquement créées lorsque vous utilisez la fédération ou que vous changez de rôle. AWS recommande de générer dynamiquement des informations d'identification temporaires au lieu d'utiliser des clés d'accès à long terme. Pour plus d'informations, consultez Informations d'identification de sécurité temporaires dans IAM et Services AWS compatibles avec IAM dans le Guide de l'utilisateur IAM.

Transfert des sessions d'accès pour Amazon Managed Service for Prometheus

Prend en charge les sessions d'accès direct (FAS) : oui

Les sessions d'accès direct (FAS) utilisent les autorisations du principal appelant et Service AWS, combinées Service AWS à la demande d'envoi de demandes aux services en aval. Pour plus de détails sur les politiques relatives à l'envoi de demandes FAS, consultez la section Transférer les sessions d'accès.

Rôles de service pour Amazon Managed Service for Prometheus

Prend en charge les rôles de service : Non

Un rôle de service est un rôle IAM qu'un service endosse pour accomplir des actions en votre nom. Un administrateur IAM peut créer, modifier et supprimer un rôle de service à partir d'IAM. Pour plus d'informations, consultez Création d'un rôle pour la délégation d'autorisations à un Service AWS dans le Guide de l'utilisateur IAM.

> **Warning**
>
> La modification des autorisations d'un rôle de service peut altérer la fonctionnalité d'Amazon Managed Service for Prometheus. Ne modifiez des rôles de service que quand Amazon Managed Service for Prometheus vous le conseille.

Rôles liés à un service pour Amazon Managed Service for Prometheus

Prend en charge les rôles liés à un service : oui

Un rôle lié à un service est un type de rôle de service lié à un. Service AWS Le service peut endosser le rôle afin d'effectuer une action en votre nom. Les rôles liés à un service apparaissent dans votre Compte AWS répertoire et appartiennent au service. Un administrateur IAM peut consulter, mais ne peut pas modifier, les autorisations concernant les rôles liés à un service.

Pour plus d'informations sur la création ou la gestion de rôles liés à un service Amazon Managed Service for Prometheus, consultez la section Utilisation de rôles liés à un service pour Amazon Managed Service for Prometheus.

Exemples de politiques basées sur l'identité pour Amazon Managed Service for Prometheus

Par défaut, les utilisateurs et les rôles ne sont pas autorisés à créer ou à modifier des ressources Amazon Managed Service for Prometheus. Pour octroyer aux utilisateurs des autorisations d'effectuer des actions sur les ressources dont ils ont besoin, un administrateur IAM peut créer des politiques IAM.

Pour apprendre à créer une politique basée sur l'identité IAM à l'aide de ces exemples de documents de politique JSON, consultez Création de politiques IAM (console) dans le Guide de l'utilisateur IAM.

Pour plus de détails sur les actions et les types de ressources définis par Amazon Managed Service pour Prometheus, y compris le format du ARNs pour chacun des types de ressources, consultez Actions, ressources et clés de condition pour Amazon Managed Service for Prometheus dans la référence d'autorisation du service.

Rubriques

- Bonnes pratiques en matière de politiques
- Utilisation de la console Amazon Managed Service for Prometheus
- Autorisation accordée aux utilisateurs pour afficher leurs propres autorisations

Bonnes pratiques en matière de politiques

Les politiques basées sur l'identité déterminent si une personne peut créer, consulter ou supprimer des ressources Amazon Managed Service for Prometheus dans votre compte. Ces actions peuvent entraîner des frais pour votre Compte AWS. Lorsque vous créez ou modifiez des politiques basées sur l'identité, suivez ces instructions et recommandations :

- Commencez AWS par les politiques gérées et passez aux autorisations du moindre privilège : pour commencer à accorder des autorisations à vos utilisateurs et à vos charges de travail, utilisez les politiques AWS gérées qui accordent des autorisations pour de nombreux cas d'utilisation courants. Ils sont disponibles dans votre Compte AWS. Nous vous recommandons de réduire davantage les autorisations en définissant des politiques gérées par les AWS clients spécifiques à

vos cas d'utilisation. Pour plus d'informations, consultez politiques gérées par AWS ou politiques gérées par AWS pour les activités professionnelles dans le Guide de l'utilisateur IAM.

- Accordez les autorisations de moindre privilège : lorsque vous définissez des autorisations avec des politiques IAM, accordez uniquement les autorisations nécessaires à l'exécution d'une seule tâche. Pour ce faire, vous définissez les actions qui peuvent être entreprises sur des ressources spécifiques dans des conditions spécifiques, également appelées autorisations de moindre privilège. Pour plus d'informations sur l'utilisation d'IAM pour appliquer des autorisations, consultez politiques et autorisations dans IAM dans le Guide de l'utilisateur IAM.
- Utilisez des conditions dans les politiques IAM pour restreindre davantage l'accès : vous pouvez ajouter une condition à vos politiques afin de limiter l'accès aux actions et aux ressources. Par exemple, vous pouvez écrire une condition de politique pour spécifier que toutes les demandes doivent être envoyées via SSL. Vous pouvez également utiliser des conditions pour accorder l'accès aux actions de service si elles sont utilisées par le biais d'un service spécifique Service AWS, tel que CloudFormation. Pour plus d'informations, consultez Conditions pour éléments de politique JSON IAM dans le Guide de l'utilisateur IAM.
- Utilisez l'Analyseur d'accès IAM pour valider vos politiques IAM afin de garantir des autorisations sécurisées et fonctionnelles : l'Analyseur d'accès IAM valide les politiques nouvelles et existantes de manière à ce que les politiques IAM respectent le langage de politique IAM (JSON) et les bonnes pratiques IAM. IAM Access Analyzer fournit plus de 100 vérifications de politiques et des recommandations exploitables pour vous aider à créer des politiques sécurisées et fonctionnelles. Pour plus d'informations, consultez Validation de politiques avec IAM Access Analyzer dans le Guide de l'utilisateur IAM.
- Exiger l'authentification multifactorielle (MFA) : si vous avez un scénario qui nécessite des utilisateurs IAM ou un utilisateur root, activez l'authentification MFA pour une sécurité accrue. Compte AWS Pour exiger la MFA lorsque des opérations d'API sont appelées, ajoutez des conditions MFA à vos politiques. Pour plus d'informations, consultez Sécurisation de l'accès aux API avec MFA dans le Guide de l'utilisateur IAM.

Pour plus d'informations sur les bonnes pratiques dans IAM, consultez Bonnes pratiques de sécurité dans IAM dans le Guide de l'utilisateur IAM.

Utilisation de la console Amazon Managed Service for Prometheus

Pour accéder à la console Amazon Managed Service for Prometheus, vous devez disposer d'un ensemble minimum d'autorisations. Ces autorisations doivent vous permettre de répertorier et de consulter des informations sur les ressources Amazon Managed Service for Prometheus dans votre

Compte AWS. Si vous créez une politique basée sur l'identité qui est plus restrictive que l'ensemble minimum d'autorisations requis, la console ne fonctionnera pas comme prévu pour les entités (utilisateurs ou rôles) tributaires de cette politique.

Il n'est pas nécessaire d'accorder des autorisations de console minimales aux utilisateurs qui appellent uniquement l'API AWS CLI ou l' AWS API. Autorisez plutôt l'accès à uniquement aux actions qui correspondent à l'opération d'API qu'ils tentent d'effectuer.

Pour garantir que les utilisateurs et les rôles peuvent toujours utiliser la console Amazon Managed Service for Prometheus, associez également l'Amazon Managed Service for Prometheus ou la politique gérée aux `ConsoleAccess` entités `ReadOnly` AWS . Pour plus d'informations, consultez Ajout d'autorisations à un utilisateur dans le Guide de l'utilisateur IAM.

Autorisation accordée aux utilisateurs pour afficher leurs propres autorisations

Cet exemple montre comment créer une politique qui permet aux utilisateurs IAM d'afficher les politiques en ligne et gérées attachées à leur identité d'utilisateur. Cette politique inclut les autorisations permettant d'effectuer cette action sur la console ou par programmation à l'aide de l'API AWS CLI or AWS .

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
```

```
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

Résolution des problèmes liés à l'identité et aux accès dans Amazon Managed Service for Prometheus

Utilisez les informations suivantes pour identifier et résoudre les problèmes courants que vous pouvez rencontrer lorsque vous utilisez Amazon Managed Service for Prometheus et IAM.

Rubriques

- Je ne suis pas autorisé à effectuer une action dans Amazon Managed Service for Prometheus
- Je ne suis pas autorisé à effectuer iam : PassRole
- Je souhaite autoriser des personnes extérieures à mon AWS compte à accéder à mes ressources Amazon Managed Service for Prometheus

Je ne suis pas autorisé à effectuer une action dans Amazon Managed Service for Prometheus

Si vous recevez une erreur qui indique que vous n'êtes pas autorisé à effectuer une action, vos politiques doivent être mises à jour afin de vous permettre d'effectuer l'action.

L'exemple d'erreur suivant se produit quand l'utilisateur IAM `mateojackson` tente d'utiliser la console pour afficher des informations détaillées sur une ressource *`my-example-widget`* fictive, mais ne dispose pas des autorisations `aps:`*`GetWidget`* fictives.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 aps:GetWidget on resource: my-example-widget
```

Dans ce cas, la politique qui s'applique à l'utilisateur `mateojackson` doit être mise à jour pour autoriser l'accès à la ressource *`my-example-widget`* à l'aide de l'action `aps:GetWidget`.

Si vous avez besoin d'aide, contactez votre AWS administrateur. Votre administrateur vous a fourni vos informations d'identification de connexion.

Je ne suis pas autorisé à effectuer iam : PassRole

Si vous recevez une erreur selon laquelle vous n'êtes pas autorisé à exécuter l'action `iam:PassRole`, vos politiques doivent être mises à jour pour vous permettre de transmettre un rôle à Amazon Managed Service for Prometheus.

Certains vous Services AWS permettent de transmettre un rôle existant à ce service au lieu de créer un nouveau rôle de service ou un rôle lié à un service. Pour ce faire, vous devez disposer des autorisations nécessaires pour transmettre le rôle au service.

L'exemple d'erreur suivant se produit lorsqu'un utilisateur IAM nommé `marymajor` essaie d'utiliser la console pour effectuer une action dans Amazon Managed Service for Prometheus. Toutefois, l'action nécessite que le service ait des autorisations accordées par un rôle de service. Mary n'est pas autorisée à transmettre le rôle au service.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

Dans ce cas, les politiques de Mary doivent être mises à jour pour lui permettre d'exécuter l'action `iam:PassRole`.

Si vous avez besoin d'aide, contactez votre AWS administrateur. Votre administrateur vous a fourni vos informations d'identification de connexion.

Je souhaite autoriser des personnes extérieures à mon AWS compte à accéder à mes ressources Amazon Managed Service for Prometheus

Vous pouvez créer un rôle que les utilisateurs provenant d'autres comptes ou les personnes extérieures à votre organisation pourront utiliser pour accéder à vos ressources. Vous pouvez spécifier qui est autorisé à assumer le rôle. Pour les services qui prennent en charge les politiques basées sur les ressources ou les listes de contrôle d'accès (ACLs), vous pouvez utiliser ces politiques pour autoriser les utilisateurs à accéder à vos ressources.

Pour plus d'informations, consultez les éléments suivants :

- Pour savoir si Amazon Managed Service for Prometheus prend en charge ces fonctionnalités, consultez la section Utilisation d'Amazon Managed Service for Prometheus avec IAM.
- Pour savoir comment fournir l'accès à vos ressources sur celles Comptes AWS que vous possédez, consultez la section Fournir l'accès à un utilisateur IAM dans un autre utilisateur Compte AWS que vous possédez dans le Guide de l'utilisateur IAM.
- Pour savoir comment fournir l'accès à vos ressources à des tiers Comptes AWS, consultez la section Fournir un accès à des ressources Comptes AWS détenues par des tiers dans le guide de l'utilisateur IAM.
- Pour savoir comment fournir un accès par le biais de la fédération d'identité, consultez Fournir un accès à des utilisateurs authentifiés en externe (fédération d'identité) dans le Guide de l'utilisateur IAM.
- Pour en savoir plus sur la différence entre l'utilisation des rôles et des politiques basées sur les ressources pour l'accès intercompte, consultez Accès intercompte aux ressources dans IAM dans le Guide de l'utilisateur IAM.

Autorisations et politiques IAM

L'accès aux actions et aux données Amazon Managed Service for Prometheus nécessite des informations d'identification. Ces informations d'identification doivent être autorisées à effectuer les actions et à accéder aux AWS ressources, telles que la récupération des données Amazon Managed Service for Prometheus concernant vos ressources cloud. Les sections suivantes fournissent des informations sur la manière dont vous pouvez utiliser Gestion des identités et des accès AWS (IAM) et Amazon Managed Service for Prometheus pour sécuriser vos ressources, en contrôlant qui peut y accéder. Pour plus d'informations, consultez la section Policies and permissions in IAM.

Autorisations Amazon Managed Service for Prometheus

Pour consulter la liste des actions possibles d'Amazon Managed Service for Prometheus, des types de ressources et des clés de condition, consultez Actions, ressources et clés de condition pour Amazon Managed Service for Prometheus.

Exemple de politiques IAM

Cette section fournit des exemples d'autres politiques autogérées que vous pouvez créer.

La politique IAM suivante accorde un accès complet à Amazon Managed Service for Prometheus et permet également à un utilisateur de découvrir les clusters Amazon EKS et de consulter les détails les concernant.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "aps:*",
                "eks:DescribeCluster",
                "eks:ListClusters"
            ],
            "Resource": "*"
        }
    ]
}
```

Validation de la conformité pour Amazon Managed Service for Prometheus

Notre nouvelle expérience AWS d'inscription n'est pas conçue pour les charges de travail réglementées. Si vous utilisez notre nouvelle interface d' AWS inscription, mais que vous souhaitez l'utiliser AWS pour des charges de travail réglementées, vous pouvez vous inscrire à des fonctionnalités AWS (avancées) ou activer des fonctionnalités avancées pour votre AWS environnement.

Pour savoir si un s' Service AWS inscrit dans le cadre de programmes de conformité spécifiques, consultez Services AWS la section Étendue par programme de conformité et choisissez le programme de conformité qui vous intéresse. Pour des informations générales, voir Programmes de AWS conformité Programmes .

Vous pouvez télécharger des rapports d'audit tiers à l'aide de AWS Artifact. Pour plus d'informations, voir Téléchargement de rapports dans AWS Artifact .

Votre responsabilité en matière de conformité lors de l'utilisation Services AWS dépend de la sensibilité de vos données, des objectifs de conformité de votre entreprise et des lois et réglementations applicables. Pour plus d'informations sur votre responsabilité en matière de conformité lors de l'utilisation Services AWS, consultez AWS la documentation de sécurité.

Résilience dans Amazon Managed Service for Prometheus

L'infrastructure AWS mondiale est construite autour des AWS régions et des zones de disponibilité. AWS Les régions fournissent plusieurs zones de disponibilité physiquement séparées et isolées, connectées par un réseau à faible latence, à haut débit et hautement redondant. Avec les zones de disponibilité, vous pouvez concevoir et exploiter des applications et des bases de données qui basculent automatiquement d'une zone à l'autre sans interruption. Les zones de disponibilité sont davantage disponibles, tolérantes aux pannes et ont une plus grande capacité de mise à l'échelle que les infrastructures traditionnelles à un ou plusieurs centres de données.

Pour plus d'informations sur AWS les régions et les zones de disponibilité, consultez la section Infrastructure AWS mondiale.

Outre l'infrastructure AWS mondiale, Amazon Managed Service for Prometheus propose plusieurs fonctionnalités pour répondre à vos besoins en matière de résilience et de sauvegarde des données, notamment la prise en charge des données à haute disponibilité.

Sécurité de l'infrastructure dans Amazon Managed Service for Prometheus

En tant que service géré, Amazon Managed Service for Prometheus est protégé AWS par la sécurité du réseau mondial. Pour plus d'informations sur les services AWS de sécurité et sur la manière dont AWS l'infrastructure est protégée, consultez la section Sécurité du AWS cloud. Pour concevoir votre AWS environnement en utilisant les meilleures pratiques en matière de sécurité de l'infrastructure, consultez la section Protection de l'infrastructure dans le cadre AWS bien architecturé du pilier de sécurité.

Vous utilisez des appels d'API AWS publiés pour accéder à Amazon Managed Service for Prometheus via le réseau. Les clients doivent prendre en charge les éléments suivants :

- Protocole TLS (Transport Layer Security). Nous exigeons TLS 1.2 et recommandons TLS 1.3.

- Ses suites de chiffrement PFS (Perfect Forward Secrecy) comme DHE (Ephemeral Diffie-Hellman) ou ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). La plupart des systèmes modernes tels que Java 7 et les versions ultérieures prennent en charge ces modes.

Utilisation de rôles liés à un service pour Amazon Managed Service for Prometheus

Amazon Managed Service pour les utilisations de Prometheus Gestion des identités et des accès AWS (IAM) rôles liés aux services. Un rôle lié à un service est un type unique de rôle IAM directement lié à Amazon Managed Service pour Prometheus. Service-linked les rôles sont prédéfinis par Amazon Managed Service pour Prometheus et incluent toutes les autorisations dont le service a besoin pour appeler d'autres AWS services en votre nom.

Un rôle lié à un service simplifie la configuration d'Amazon Managed Service for Prometheus, car vous n'avez pas besoin d'ajouter manuellement les autorisations requises. Amazon Managed Service for Prometheus définit les autorisations de ses rôles liés à un service ; sauf définition contraire, seul Amazon Managed Service for Prometheus peut endosser ses rôles. Les autorisations définies comprennent la politique d'approbation et la politique d'autorisation. De plus, cette politique d'autorisation ne peut pas être attachée à une autre entité IAM.

Utilisation de rôles pour récupérer des métriques d'EKS

Lorsque vous extrayez automatiquement des métriques à l'aide du collecteur géré Amazon Managed Service pour Prometheus, le AWSServiceRoleForAmazonPrometheusScraper le rôle lié à un service est utilisé pour faciliter la configuration de Managed Collector, car vous n'avez pas à ajouter manuellement les autorisations nécessaires. Amazon Managed Service for Prometheus définit les autorisations et peut, seul, endosser le rôle.

Pour plus d'informations sur les autres services qui prennent en charge les rôles liés aux services, consultez AWS services qui fonctionnent avec IAM et recherchez les services qui ont Oui dans le Service-linked rôles colonne. Choisissez un Oui ayant un lien permettant de consulter les détails du rôle pour ce service.

Service-linked autorisations de rôle pour Amazon Managed Service pour Prometheus

Amazon Managed Service pour Prometheus utilise un rôle lié à un service nommé avec le préfixe AWSServiceRoleForAmazonPrometheusScraper pour permettre à Amazon Managed Service for Prometheus de récupérer automatiquement les métriques de vos clusters Amazon EKS.

Le AWSServiceRoleForAmazonPrometheusScraper le rôle lié à un service fait confiance aux services suivants pour assumer le rôle :

- `scraper.aps.amazonaws.com`

La politique d'autorisations de rôle nommée AmazonPrometheusScraperServiceRolePolicy permet à Amazon Managed Service for Prometheus d'effectuer les actions suivantes sur les ressources spécifiées :

- Lisez et modifiez la configuration réseau pour vous connecter au réseau qui contient votre cluster Amazon EKS.
- Lire les métriques des clusters Amazon EKS et les inscrire dans vos espaces de travail Amazon Managed Service for Prometheus.
- Écrire des métriques à Amazon CloudWatch Ensembles de données de métriques dans le même compte.

Le rôle ne dispose d'aucune autorisation Amazon MSK par défaut. Pour collecter des statistiques sur les groupes de consommateurs à l'aide d'un exportateur Kafka géré, accordez l'accès au rôle avec une politique de cluster Amazon MSK. Pour plus d'informations, voir Autoriser le scraper à accéder au cluster.

Vous devez configurer les autorisations pour permettre à vos utilisateurs, groupes ou rôles de créer un rôle lié à un service. Pour plus d'informations, voir Service-linked autorisations de rôle dans le Guide de l'utilisateur IAM.

Création d'un rôle lié à un service pour Amazon Managed Service for Prometheus

Vous n'avez pas besoin de créer manuellement un rôle lié à un service. Lorsque vous créez une instance de collecteur gérée à l'aide d'Amazon EKS ou d'Amazon Managed Service pour Prometheus dans Console de gestion AWS, le AWS CLI, ou le AWS API, Amazon Managed Service for Prometheus crée le rôle lié au service pour vous.

> **Important**
>
> Ce rôle lié à un service peut apparaître dans votre compte si vous avez effectué une action dans un autre service qui utilise les fonctions prises en charge par ce rôle. Pour en savoir plus, consultez Un nouveau rôle est apparu dans mon Compte AWS.

Si vous supprimez ce rôle lié à un service et que vous avez ensuite besoin de le recréer, vous pouvez utiliser la même procédure pour recréer le rôle dans votre compte. Lorsque vous créez une instance de collecteur géré à l'aide d'Amazon EKS ou d'Amazon Managed Service for Prometheus, Amazon Managed Service for Prometheus crée à nouveau le rôle lié à un service pour vous.

Modification d'un rôle lié à un service pour Amazon Managed Service for Prometheus

Amazon Managed Service pour Prometheus ne vous permet pas de modifier le AWSServiceRoleForAmazonPrometheusScraper rôle lié à un service. Après avoir créé un rôle lié à un service, vous ne pouvez pas changer le nom du rôle, car plusieurs entités peuvent faire référence à ce rôle. Néanmoins, vous pouvez modifier la description du rôle à l'aide d'IAM. Pour en savoir plus, consultez Modification d'un rôle lié à un service dans le guide de l'utilisateur IAM.

Suppression d'un rôle lié à un service d'Amazon Managed Service for Prometheus

Il n'est pas nécessaire de supprimer manuellement le AWSServiceRoleForAmazonPrometheusScraper rôle. Lorsque vous supprimez toutes les instances de collecteur gérées associées au rôle dans Console de gestion AWS, le AWS CLI, ou le AWS L'API Amazon Managed Service for Prometheus nettoie les ressources et supprime le rôle lié au service pour vous.

Régions prises en charge pour les rôles liés à un service d'Amazon Managed Service for Prometheus

Amazon Managed Service for Prometheus prend en charge l'utilisation des rôles liés à un service dans toutes les régions dans lesquelles le service est disponible. Pour plus d'informations, consultez Régions prises en charge.

Journalisation des appels d'API Amazon Managed Service pour Prometheus à l'aide de AWS CloudTrail

Amazon Managed Service for Prometheus est intégré AWS CloudTrailà un service qui fournit un enregistrement des actions entreprises par un utilisateur, un rôle ou un. Service AWS CloudTrailcapture tous les appels d'API pour Amazon Managed Service for Prometheus sous forme d'événements. Les appels capturés incluent des appels provenant de la console Amazon Managed Service for Prometheus et des appels de code vers Amazon Managed Service pour les opérations de l'API Prometheus. À l'aide des informations collectées par CloudTrail, vous pouvez déterminer la

demande envoyée à Amazon Managed Service pour Prometheus, l'adresse IP à partir de laquelle la demande a été faite, la date à laquelle elle a été faite et des informations supplémentaires.

Chaque événement ou entrée de journal contient des informations sur la personne ayant initié la demande. Les informations relatives à l'identité permettent de déterminer :

- Si la demande a été effectuée avec des informations d'identification d'utilisateur root ou d'utilisateur root.
- Si la demande a été faite au nom d'un utilisateur du centre d'identité IAM.
- Si la demande a été effectuée avec les informations d'identification de sécurité temporaires d'un rôle ou d'un utilisateur fédéré.
- Si la requête a été effectuée par un autre Service AWS.

CloudTrail est actif dans votre compte Compte AWS lorsque vous créez le compte et vous avez automatiquement accès à l'historique des CloudTrail événements. L'historique des CloudTrail événements fournit un enregistrement consultable, consultable, téléchargeable et immuable des 90 derniers jours des événements de gestion enregistrés dans un. Région AWS Pour plus d'informations, consultez la section Utilisation de l'historique des CloudTrail événements dans le guide de AWS CloudTrail l'utilisateur. La consultation de CloudTrail l'historique des événements est gratuite.

Pour un enregistrement continu des événements de vos 90 Compte AWS derniers jours, créez un magasin de données sur les événements de Trail ou CloudTrail Lake.

CloudTrail sentiers

Un suivi permet CloudTrail de fournir des fichiers journaux à un compartiment Amazon S3. Tous les sentiers créés à l'aide du Console de gestion AWS sont multirégionaux. Vous ne pouvez créer un journal de suivi en une ou plusieurs régions à l'aide de l' AWS CLI. Il est recommandé de créer un parcours multirégional, car vous capturez l'activité dans l'ensemble Régions AWS de votre compte. Si vous créez un journal de suivi pour une seule région, il convient de n'afficher que les événements enregistrés dans le journal de suivi pour une seule région Région AWS. Pour plus d'informations sur les journaux de suivi, consultez Créez un journal de suivi dans vos Compte AWS et Création d'un journal de suivi pour une organisation dans le AWS CloudTrail Guide de l'utilisateur.

Vous pouvez envoyer une copie de vos événements de gestion en cours dans votre compartiment Amazon S3 gratuitement CloudTrail en créant un journal. Toutefois, des frais de stockage

Amazon S3 sont facturés. Pour plus d'informations sur la CloudTrail tarification, consultez la section AWS CloudTrail Tarification. Pour obtenir des informations sur la tarification Amazon S3, consultez Tarification Amazon S3.

CloudTrail Stockages de données sur les événements du lac

CloudTrail Lake vous permet d'exécuter des requêtes SQL sur vos événements. CloudTrail Lake convertit les événements existants au format JSON basé sur les lignes au format Apache ORC. ORC est un format de stockage en colonnes qui est optimisé pour une récupération rapide des données. Les événements sont agrégés dans des magasins de données d'événement. Ceux-ci constituent des collections immuables d'événements basées sur des critères que vous sélectionnez en appliquant des sélecteurs d'événements avancés. Les sélecteurs que vous appliquez à un magasin de données d'événement contrôlent les événements qui persistent et que vous pouvez interroger. Pour plus d'informations sur CloudTrail Lake, consultez la section Travailler avec AWS CloudTrail Lake dans le guide de AWS CloudTrail l'utilisateur.

CloudTrail Les stockages et requêtes de données sur les événements de Lake entraînent des coûts. Lorsque vous créez un magasin de données d'événement, vous choisissez l'option de tarification que vous voulez utiliser pour le magasin de données d'événement. L'option de tarification détermine le coût d'ingestion et de stockage des événements, ainsi que les périodes de conservation par défaut et maximale pour le magasin de données d'événement. Pour plus d'informations sur la CloudTrail tarification, consultez la section AWS CloudTrail Tarification.

Événements de gestion d'Amazon Managed Service for Prometheus dans CloudTrail

Les événements de gestion fournissent des informations sur les opérations de gestion effectuées sur les ressources de votre Compte AWS. Ils sont également connus sous le nom opérations de plan de contrôle. Par défaut, CloudTrail enregistre les événements de gestion.

Amazon Managed Service for Prometheus enregistre toutes les opérations du plan de contrôle Amazon Managed Service for Prometheus en tant qu'événements de gestion. Pour obtenir la liste des opérations du plan de contrôle Amazon Managed Service for Prometheus auxquelles Amazon Managed Service for Prometheus se connecte CloudTrail, consultez le manuel Amazon Managed Service for Prometheus API Reference.

Exemples d'événements Amazon Managed Service pour Prometheus

Un événement représente une demande unique provenant de n'importe quelle source et inclut des informations sur l'opération d'API demandée, la date et l'heure de l'opération, les paramètres de la demande, etc. CloudTrail les fichiers journaux ne constituent pas une trace ordonnée des appels d'API publics. Les événements n'apparaissent donc pas dans un ordre spécifique.

Exemple : CreateWorkspace

L'exemple suivant montre une entrée de CloudTrail journal illustrant l' CreateWorkspaceaction.


```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "EXAMPLE123EXAMPLE123-1234567890616",
        "arn": "arn:aws:sts::123456789012:assumed-role/Admin/admin",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                "arn": "arn:aws:iam::123456789012:role/Admin",
                "accountId": "123456789012",
                "userName": "Admin"
            },
            "webIdFederationData": {

            },
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2020-11-30T23:39:29Z"
            }
        }
    },
    "eventTime": "2020-11-30T23:43:21Z",
    "eventSource": "aps.amazonaws.com",
    "eventName": "CreateWorkspace",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "203.0.113.1",
    "userAgent": "aws-cli/1.11.167 Python/2.7.10 Darwin/16.7.0 botocore/1.7.25",
    "requestParameters": {
```

```
        "alias": "alias-example",
        "clientToken": "12345678-1234-abcd-1234-12345abcd1"
    },
    "responseElements": {
        "Access-Control-Expose-Headers": "x-amzn-errortype,x-amzn-requestid,x-amzn-
trace-id,x-amzn-errormessage,x-amz-apigw-id,date",
        "arn": "arn:aws:aps:us-west-2:123456789012:workspace/ws-abc123456-
abcd-1234-5678-1234567890",
        "status": {
            "statusCode": "CREATING"
        },
        "workspaceId": "ws-12345678-1234-abcd-1234-1234567890"
    },
    "requestID": "890b8639-e51f-11e7-b038-EXAMPLE",
    "eventID": "874f89fa-70fc-4798-bc00-EXAMPLE",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "eventCategory": "Management",
    "recipientAccountId": "123456789012"
}
```

Exemple : CreateAlertManagerDefinition

L'exemple suivant montre une entrée de CloudTrail journal illustrant l' CreateAlertManagerDefinition action.

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "EXAMPLE123EXAMPLE123-1234567890616",
        "arn": "arn:aws:sts::123456789012:assumed-role/Admin/admin",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                "arn": "arn:aws:iam::123456789012:role/Admin",
                "accountId": "123456789012",
                "userName": "Admin"
            },
```

```
            "webIdFederationData": {

            },
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2021-09-23T20:20:14Z"
            }
        }
    },
    "eventTime": "2021-09-23T20:22:43Z",
    "eventSource": "aps.amazonaws.com",
    "eventName": "CreateAlertManagerDefinition",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "203.0.113.1",
    "userAgent": "Boto3/1.17.46 Python/3.6.14 Linux/4.14.238-182.422.amzn2.x86_64 exec-
env/AWS_ECS_FARGATE Botocore/1.20.46",
    "requestParameters": {
        "data":
 "YWxlcnRtYW5hZ2VyX2NvbmZpZzogfAogIGdsb2JhbDoKICAgIHNtdHBfc21hcnRob3N0OiAnbG9jYWxob3N0OjI1JwogI
        "clientToken": "12345678-1234-abcd-1234-12345abcd1",
        "workspaceId": "ws-12345678-1234-abcd-1234-1234567890"
    },
    "responseElements": {
        "Access-Control-Expose-Headers": "x-amzn-errortype,x-amzn-requestid,x-amzn-
trace-id,x-amzn-errormessage,x-amz-apigw-id,date",
        "status": {
            "statusCode": "CREATING"
        }
    },
    "requestID": "890b8639-e51f-11e7-b038-EXAMPLE",
    "eventID": "874f89fa-70fc-4798-bc00-EXAMPLE",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "eventCategory": "Management",
    "recipientAccountId": "123456789012"
}
```


Exemple : CreateRuleGroupsNamespace

L'exemple suivant montre une entrée de CloudTrail journal illustrant l' CreateRuleGroupsNamespace action.

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "EXAMPLE123EXAMPLE123-1234567890616",
        "arn": "arn:aws:sts::123456789012:assumed-role/Admin/admin",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                "arn": "arn:aws:iam::123456789012:role/Admin",
                "accountId": "123456789012",
                "userName": "Admin"
            },
            "webIdFederationData": {

            },
            "attributes": {
                "creationDate": "2021-09-23T20:22:19Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2021-09-23T20:25:08Z",
    "eventSource": "aps.amazonaws.com",
    "eventName": "CreateRuleGroupsNamespace",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "34.212.33.165",
    "userAgent": "Boto3/1.17.63 Python/3.6.14 Linux/4.14.238-182.422.amzn2.x86_64 exec-
env/AWS_ECS_FARGATE Botocore/1.20.63",
    "requestParameters": {
        "data":
 "Z3JvdXBzOgogIC0gbmFtZTogdGVzdFJ1bGVHcm91cHNOYW1lc3BhY2UKICAgIHJ1bGVzOgogICAgLSBhbGVydDogdGVzd
        "clientToken": "12345678-1234-abcd-1234-12345abcd1",
        "name": "exampleRuleGroupsNamespace",
        "workspaceId": "ws-12345678-1234-abcd-1234-1234567890"
    },
    "responseElements": {
        "Access-Control-Expose-Headers": "x-amzn-errortype,x-amzn-requestid,x-amzn-
trace-id,x-amzn-errormessage,x-amz-apigw-id,date",
        "name": "exampleRuleGroupsNamespace",
```

```
        "arn": "arn:aws:aps:us-west-2:492980759322:rulegroupsnamespace/ws-
ae46a85c-1609-4c22-90a3-2148642c3b6c/exampleRuleGroupsNamespace",
        "status": {
            "statusCode": "CREATING"
        },
        "tags": {}
    },
    "requestID": "890b8639-e51f-11e7-b038-EXAMPLE",
    "eventID": "874f89fa-70fc-4798-bc00-EXAMPLE",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "eventCategory": "Management",
    "recipientAccountId": "123456789012"
}
```


Pour plus d'informations sur le contenu des CloudTrail enregistrements, voir le contenu des CloudTrail enregistrements dans le Guide de AWS CloudTrail l'utilisateur.

Configuration de rôles IAM pour les comptes de service

Avec les rôles IAM pour les comptes de service, vous pouvez associer un rôle IAM à un compte de service Kubernetes. Ce compte de service peut ensuite fournir des AWS autorisations aux conteneurs de n'importe quel pod utilisant ce compte de service. Pour plus d'informations, consultez la section Rôles IAM pour les comptes de service.

Les rôles IAM pour les comptes de service sont également appelés rôles de service.

Dans Amazon Managed Service for Prometheus, l'utilisation de rôles de service permet d'obtenir les rôles dont vous avez besoin pour autoriser et authentifier sur Amazon Managed Service for Prometheus, les serveurs Prometheus et les serveurs Grafana.

Conditions préalables

Les procédures de cette page nécessitent l'installation de l'interface de ligne de commande AWS CLI et EKSCTL.

Configuration de rôles de service pour l'ingestion de métriques à partir de clusters Amazon EKS

Pour configurer les rôles de service afin de permettre à Amazon Managed Service for Prometheus d'ingérer les métriques des serveurs Prometheus dans les clusters Amazon EKS, vous devez être connecté à un compte et disposer des autorisations suivantes :

- `iam:CreateRole`
- `iam:CreatePolicy`
- `iam:GetRole`
- `iam:AttachRolePolicy`
- `iam:GetOpenIDConnectProvider`

Pour configurer le rôle de service pour l'ingestion dans Amazon Managed Service for Prometheus

1. Créez un fichier nommé `createIRSA-AMPIngest.sh` avec le contenu suivant. Remplacez `<my_amazon_eks_clustername>` par le nom de votre cluster et `<my_prometheus_namespace>` par votre espace de noms Prometheus.

```
#!/bin/bash -e
CLUSTER_NAME=<my_amazon_eks_clustername>
SERVICE_ACCOUNT_NAMESPACE=<my_prometheus_namespace>
AWS_ACCOUNT_ID=$(aws sts get-caller-identity --query "Account" --output text)
OIDC_PROVIDER=$(aws eks describe-cluster --name $CLUSTER_NAME --query
 "cluster.identity.oidc.issuer" --output text | sed -e "s/^https:\/\///")
SERVICE_ACCOUNT_AMP_INGEST_NAME=amp-iamproxy-ingest-service-account
SERVICE_ACCOUNT_IAM_AMP_INGEST_ROLE=amp-iamproxy-ingest-role
SERVICE_ACCOUNT_IAM_AMP_INGEST_POLICY=AMPIngestPolicy
#
# Set up a trust policy designed for a specific combination of K8s service account
 and namespace to sign in from a Kubernetes cluster which hosts the OIDC Idp.
#
cat <<EOF > TrustPolicy.json
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
```

```
        "Federated": "arn:aws:iam::${AWS_ACCOUNT_ID}:oidc-provider/
${OIDC_PROVIDER}"
      },
      "Action": "sts:AssumeRoleWithWebIdentity",
      "Condition": {
        "StringEquals": {
          "${OIDC_PROVIDER}:sub": "system:serviceaccount:
${SERVICE_ACCOUNT_NAMESPACE}:${SERVICE_ACCOUNT_AMP_INGEST_NAME}"
        }
      }
    }
  ]
}
EOF
#
# Set up the permission policy that grants ingest (remote write) permissions for
 all AMP workspaces
#
cat <<EOF > PermissionPolicyIngest.json
{
  "Version": "2012-10-17",
   "Statement": [
       {"Effect": "Allow",
        "Action": [
           "aps:RemoteWrite",
           "aps:GetSeries",
           "aps:GetLabels",
           "aps:GetMetricMetadata"
        ],
        "Resource": "*"
      }
   ]
}
EOF

function getRoleArn() {
  OUTPUT=$(aws iam get-role --role-name $1 --query 'Role.Arn' --output text 2>&1)

  # Check for an expected exception
  if [[ $? -eq 0 ]]; then
    echo $OUTPUT
  elif [[ -n $(grep "NoSuchEntity" <<< $OUTPUT) ]]; then
    echo ""
  else
```

```
    >&2 echo $OUTPUT
    return 1
  fi
}

#
# Create the IAM Role for ingest with the above trust policy
#
SERVICE_ACCOUNT_IAM_AMP_INGEST_ROLE_ARN=$(getRoleArn
 $SERVICE_ACCOUNT_IAM_AMP_INGEST_ROLE)
if [ "$SERVICE_ACCOUNT_IAM_AMP_INGEST_ROLE_ARN" = "" ];
then
  #
  # Create the IAM role for service account
  #
  SERVICE_ACCOUNT_IAM_AMP_INGEST_ROLE_ARN=$(aws iam create-role \
  --role-name $SERVICE_ACCOUNT_IAM_AMP_INGEST_ROLE \
  --assume-role-policy-document file://TrustPolicy.json \
  --query "Role.Arn" --output text)
  #
  # Create an IAM permission policy
  #
  SERVICE_ACCOUNT_IAM_AMP_INGEST_ARN=$(aws iam create-policy --policy-name
 $SERVICE_ACCOUNT_IAM_AMP_INGEST_POLICY \
  --policy-document file://PermissionPolicyIngest.json \
  --query 'Policy.Arn' --output text)
  #
  # Attach the required IAM policies to the IAM role created above
  #
  aws iam attach-role-policy \
  --role-name $SERVICE_ACCOUNT_IAM_AMP_INGEST_ROLE \
  --policy-arn $SERVICE_ACCOUNT_IAM_AMP_INGEST_ARN
else
    echo "$SERVICE_ACCOUNT_IAM_AMP_INGEST_ROLE_ARN IAM role for ingest already
 exists"
fi
echo $SERVICE_ACCOUNT_IAM_AMP_INGEST_ROLE_ARN
#
# EKS cluster hosts an OIDC provider with a public discovery endpoint.
# Associate this IdP with AWS IAM so that the latter can validate and accept the
 OIDC tokens issued by Kubernetes to service accounts.
# Doing this with eksctl is the easier and best approach.
#
```

```
eksctl utils associate-iam-oidc-provider --cluster $CLUSTER_NAME --approve
```

2. Saisissez la commande suivante pour donner au script les privilèges nécessaires.

```
chmod +x createIRSA-AMPIngest.sh
```

3. Exécutez le script.

Configuration de rôles IAM de comptes de service pour l'interrogation des métriques

Pour configurer un rôle IAM pour un compte de service (rôle de service) afin de permettre l'interrogation des métriques des espaces de travail Amazon Managed Service for Prometheus, vous devez être connecté à un compte et disposer des autorisations suivantes :

- `iam:CreateRole`
- `iam:CreatePolicy`
- `iam:GetRole`
- `iam:AttachRolePolicy`
- `iam:GetOpenIDConnectProvider`

Pour configurer des rôles de service pour l'interrogation des métriques Amazon Managed Service for Prometheus

1. Créez un fichier nommé `createIRSA-AMPQuery.sh` avec le contenu suivant. Remplacez <my_amazon_eks_clustername> par le nom de votre cluster et <my_prometheus_namespace> par votre espace de noms Prometheus.

```
#!/bin/bash -e
CLUSTER_NAME=<my_amazon_eks_clustername>
SERVICE_ACCOUNT_NAMESPACE=<my_prometheus_namespace>
AWS_ACCOUNT_ID=$(aws sts get-caller-identity --query "Account" --output text)
OIDC_PROVIDER=$(aws eks describe-cluster --name $CLUSTER_NAME --query
 "cluster.identity.oidc.issuer" --output text | sed -e "s/^https:\/\///")
SERVICE_ACCOUNT_AMP_QUERY_NAME=amp-iamproxy-query-service-account
SERVICE_ACCOUNT_IAM_AMP_QUERY_ROLE=amp-iamproxy-query-role
SERVICE_ACCOUNT_IAM_AMP_QUERY_POLICY=AMPQueryPolicy
#
```

```
# Setup a trust policy designed for a specific combination of K8s service account
 and namespace to sign in from a Kubernetes cluster which hosts the OIDC Idp.
#
cat <<EOF > TrustPolicy.json
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Federated": "arn:aws:iam::${AWS_ACCOUNT_ID}:oidc-provider/
${OIDC_PROVIDER}"
      },
      "Action": "sts:AssumeRoleWithWebIdentity",
      "Condition": {
        "StringEquals": {
          "${OIDC_PROVIDER}:sub": "system:serviceaccount:
${SERVICE_ACCOUNT_NAMESPACE}:${SERVICE_ACCOUNT_AMP_QUERY_NAME}"
        }
      }
    }
  ]
}
EOF
#
# Set up the permission policy that grants query permissions for all AMP workspaces
#
cat <<EOF > PermissionPolicyQuery.json
{
  "Version": "2012-10-17",
   "Statement": [
       {"Effect": "Allow",
        "Action": [
           "aps:QueryMetrics",
           "aps:GetSeries",
           "aps:GetLabels",
           "aps:GetMetricMetadata"
        ],
        "Resource": "*"
      }
   ]
}
EOF
```

```
function getRoleArn() {
  OUTPUT=$(aws iam get-role --role-name $1 --query 'Role.Arn' --output text 2>&1)

  # Check for an expected exception
  if [[ $? -eq 0 ]]; then
    echo $OUTPUT
  elif [[ -n $(grep "NoSuchEntity" <<< $OUTPUT) ]]; then
    echo ""
  else
    >&2 echo $OUTPUT
    return 1
  fi
}

#
# Create the IAM Role for query with the above trust policy
#
SERVICE_ACCOUNT_IAM_AMP_QUERY_ROLE_ARN=$(getRoleArn
 $SERVICE_ACCOUNT_IAM_AMP_QUERY_ROLE)
if [ "$SERVICE_ACCOUNT_IAM_AMP_QUERY_ROLE_ARN" = "" ];
then
  #
  # Create the IAM role for service account
  #
  SERVICE_ACCOUNT_IAM_AMP_QUERY_ROLE_ARN=$(aws iam create-role \
  --role-name $SERVICE_ACCOUNT_IAM_AMP_QUERY_ROLE \
  --assume-role-policy-document file://TrustPolicy.json \
  --query "Role.Arn" --output text)
  #
  # Create an IAM permission policy
  #
  SERVICE_ACCOUNT_IAM_AMP_QUERY_ARN=$(aws iam create-policy --policy-name
 $SERVICE_ACCOUNT_IAM_AMP_QUERY_POLICY \
  --policy-document file://PermissionPolicyQuery.json \
  --query 'Policy.Arn' --output text)
  #
  # Attach the required IAM policies to the IAM role create above
  #
  aws iam attach-role-policy \
  --role-name $SERVICE_ACCOUNT_IAM_AMP_QUERY_ROLE \
  --policy-arn $SERVICE_ACCOUNT_IAM_AMP_QUERY_ARN
else
    echo "$SERVICE_ACCOUNT_IAM_AMP_QUERY_ROLE_ARN IAM role for query already
 exists"
```

```
fi
echo $SERVICE_ACCOUNT_IAM_AMP_QUERY_ROLE_ARN
#
# EKS cluster hosts an OIDC provider with a public discovery endpoint.
# Associate this IdP with AWS IAM so that the latter can validate and accept the
 OIDC tokens issued by Kubernetes to service accounts.
# Doing this with eksctl is the easier and best approach.
#
eksctl utils associate-iam-oidc-provider --cluster $CLUSTER_NAME --approve
```

2. Saisissez la commande suivante pour donner au script les privilèges nécessaires.

```
chmod +x createIRSA-AMPQuery.sh
```

3. Exécutez le script.

Utilisation d'Amazon Managed Service for Prometheus avec des points de terminaison de VPC d'interface

Si vous utilisez Amazon Virtual Private Cloud (Amazon VPC) pour héberger vos AWS ressources, vous pouvez établir des connexions privées entre votre VPC et Amazon Managed Service for Prometheus. Vous pouvez utiliser ces connexions pour permettre à Amazon Managed Service for Prometheus de communiquer avec vos ressources sur votre VPC sans passer par le réseau Internet public.

Amazon VPC est un AWS service que vous pouvez utiliser pour lancer AWS des ressources dans un réseau virtuel que vous définissez. Avec un VPC, vous contrôlez des paramètres réseau, tels que la plage d'adresses IP, les sous-réseaux, les tables de routage et les passerelles réseau. Pour connecter votre VPC à Amazon Managed Service for Prometheus, vous définissez un point de terminaison de VPC d'interface pour connecter votre VPC aux services AWS . Le point de terminaison assure une connectivité fiable et évolutive à Amazon Managed Service for Prometheus sans qu'une passerelle Internet, une instance NAT (Network Address Translation) ou une connexion VPN ne soit nécessaire. Pour de plus amples informations, veuillez consulter Qu'est-ce qu'Amazon VPC ? dans le Guide de l'utilisateur Amazon VPC.

Interface Les terminaux VPC sont alimentés par AWS PrivateLink une AWS technologie qui permet une communication privée entre les AWS services à l'aide d'une interface réseau élastique avec des adresses IP privées. Pour plus d'informations, consultez le billet de blog New — AWS PrivateLink for AWS Services.

Les informations suivantes sont destinés aux utilisateurs d'Amazon VPC. Pour plus d'informations sur la mise en route d'Amazon VPC, consultez la section Mise en route dans le Guide de l'utilisateur Amazon VPC.

Création d'un point de terminaison de VPC d'interface pour Amazon Managed Service for Prometheus

Créez un point de terminaison de VPC d'interface pour commencer à utiliser Amazon Managed Service for Prometheus. Choisissez parmi les points de terminaison de nom de service suivants :

- `com.amazonaws.region.aps-workspaces`

  Choisissez ce nom de service pour travailler avec Prometheus-compatible les API. Pour plus d'informations, consultez les Prometheus-compatible API dans le guide de l'utilisateur d'Amazon Managed Service for Prometheus.

- `com.amazonaws.region.aps`

  Choisissez ce nom de service pour effectuer des tâches de gestion de l'espace de travail. Pour plus d'informations, consultez la section Amazon Managed Service for Prometheus APIs dans le Guide de l'utilisateur Amazon Managed Service for Prometheus.

> **Note**
>
> Si vous utilisez remote_write dans un VPC sans accès direct à Internet, vous devez également créer une interface pour AWS Security Token Service un point de terminaison VPC, afin de permettre à sigv4 de fonctionner via le point de terminaison. Pour plus d'informations sur la création d'un point de terminaison VPC pour AWS STS, consultez la section Utilisation des points de terminaison VPC d' AWS STS interface dans le Guide de l'Gestion des identités et des accès AWS utilisateur. Vous devez configurer AWS STS pour utiliser des points de terminaison https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html régionalisés.

Pour plus d'informations, notamment des instructions étape par étape pour créer un point de terminaison de VPC d'interface, consultez la section Création d'un point de terminaison d'interface dans le Guide de l'utilisateur Amazon VPC.

> **Note**
>
> Vous pouvez utiliser les politiques de point de terminaison de VPC pour contrôler l'accès à votre point de terminaison de VPC d'interface Amazon Managed Service for Prometheus. Pour de plus amples informations, veuillez consulter la section suivante.

Si vous avez créé un point de terminaison de VPC d'interface pour Amazon Managed Service for Prometheus et que vous disposez déjà de données qui transitent vers les espaces de travail situés sur votre VPC, les métriques transiteront par le point de terminaison de VPC d'interface par défaut. Amazon Managed Service for Prometheus utilise des points de terminaison publics ou des points de terminaison d'interface privés (selon ceux utilisés) pour effectuer cette tâche.

Contrôle de l'accès à votre point de terminaison de VPC Amazon Managed Service for Prometheus

Vous pouvez utiliser les politiques de point de terminaison de VPC pour contrôler l'accès à votre point de terminaison de VPC d'interface Amazon Managed Service for Prometheus. Une stratégie de point de terminaison d'un VPC est une stratégie de ressource IAM que vous attachez à un point de terminaison lorsque vous le créez ou le modifiez. Si vous n'attachez pas de stratégie quand vous créez un point de terminaison, Amazon VPC attache une stratégie par défaut pour vous qui autorise un accès total au service. Une politique de point de terminaison n'annule pas et ne remplace pas les politiques IAM ou les politiques spécifiques aux services. Il s'agit d'une politique distincte qui contrôle l'accès depuis le point de terminaison jusqu'au service spécifié.

Pour en savoir plus, consultez Contrôle de l'accès aux services avec des points de terminaison d'un VPC dans le guide de l'utilisateur Amazon VPC.

Voici un exemple de politique de point de terminaison pour Amazon Managed Service for Prometheus. Cette politique permet aux utilisateurs ayant le rôle `PromUser`, et qui se connectent à Amazon Managed Service for Prometheus via le VPC, de voir les espaces de travail et les groupes de règles, mais pas, par exemple, de créer ou de supprimer des espaces de travail.

JSON

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

```
            "Sid": "AmazonManagedPrometheusPermissions",
            "Effect": "Allow",
            "Action": [
                "aps:DescribeWorkspace",
                "aps:DescribeRuleGroupsNamespace",
                "aps:ListRuleGroupsNamespaces",
                "aps:ListWorkspaces"
            ],
            "Resource": "arn:aws:aps:*:*:/workspaces*",
            "Principal": {
                "AWS": [
                    "arn:aws:iam::111122223333:role/PromUser"
                ]
            }
        }
    ]
}
```

L'exemple suivant montre une politique qui autorise uniquement les demandes provenant d'une adresse IP spécifiée dans le VPC spécifié. Les demandes provenant d'autres adresses IP échoueront.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": "aps:*",
            "Effect": "Allow",
            "Principal": "*",
            "Resource": "*",
            "Condition": {
                "IpAddress": {
                    "aws:VpcSourceIp": "192.0.2.123"
                },
        "StringEquals": {
                    "aws:SourceVpc": "vpc-0123456789abcdef0"
                }
            }
        }
    ]
}
```

Résoudre les erreurs liées à Amazon Managed Service for Prometheus

Utilisez les sections suivantes pour résoudre les problèmes liés à Amazon Managed Service for Prometheus.

Rubriques

- 429 ou limite d'erreurs dépassées
- Je vois des exemples en double.
- Je vois des erreurs concernant les horodatages des échantillons
- Je vois un message d'erreur lié à une limite.
- La sortie de votre serveur Prometheus local dépasse la limite.
- Certaines de mes données n'apparaissent pas
- Erreur HTTP 422 : délai d'expiration de la requête

429 ou limite d'erreurs dépassées

Si une erreur 429 similaire à l'exemple suivant s'affiche, cela signifie que vos demandes ont dépassé les quotas d'ingestion d'Amazon Managed Service for Prometheus.

```
ts=2020-10-29T15:34:41.845Z caller=dedupe.go:112 component=remote level=error
 remote_name=e13b0c
url=http://iamproxy-external.prometheus.uswest2-prod.eks:9090/workspaces/workspace_id/
api/v1/remote_write
msg="non-recoverable error" count=500 err="server returned HTTP status 429
Too Many Requests: ingestion rate limit (6666.666666666667) exceeded while adding 499
 samples and 0 metadata
```

Si une erreur 429 similaire à l'exemple suivant s'affiche, cela signifie que vos demandes ont dépassé le quota du nombre de métriques actives dans un espace de travail d'Amazon Managed Service for Prometheus.

```
ts=2020-11-05T12:40:33.375Z caller=dedupe.go:112 component=remote level=error
 remote_name=aps
```

```
url=http://iamproxy-external.prometheus.uswest2-prod.eks:9090/workspaces/workspace_id/
api/v1/remote_write
msg="non-recoverable error" count=500 err="server returned HTTP status 429 Too Many
 Requests: user=accountid_workspace_id:
per-user series limit (local limit: 0 global limit: 3000000 actual local limit: 500000)
 exceeded
```

Si vous voyez une erreur 429 similaire à l'exemple suivant, vos demandes ont dépassé le quota d'Amazon Managed Service pour Prometheus en ce qui concerne le taux (transactions par seconde) auquel vous pouvez envoyer des données à votre espace de travail à l'aide de l'API compatible `RemoteWrite` Prometheus.

```
ts=2024-03-26T16:50:21.780708811Z caller=dedupe.go:112 component=remote level=error
 remote_name=ab123c
url=https://aps-workspaces.us-east-1.amazonaws.com/workspaces/workspace_id/api/v1/
remote_write
msg="non-recoverable error" count=1000 exemplarCount=0 err="server returned HTTP status
 429 Too Many Requests: {\"message\":\"Rate exceeded\"}"
```

Si vous voyez une erreur 400 similaire à l'exemple suivant, vos demandes ont dépassé le quota d'Amazon Managed Service pour Prometheus pour les séries chronologiques actives. Pour plus de détails sur la façon dont les quotas de séries chronologiques actifs sont gérés, voirQuotas par défaut des séries actives.

```
ts=2024-03-26T16:50:21.780708811Z caller=push.go:53 level=warn
url=https://aps-workspaces.us-east-1.amazonaws.com/workspaces/workspace_id/api/v1/
remote_write
msg="non-recoverable error" count=500 exemplarCount=0
err="server returned HTTP status 400 Bad Request: maxFailure (quorum) on a given error
 family, rpc error: code = Code(400)
desc = addr=10.1.41.23:9095 state=ACTIVE zone=us-east-1a, rpc error: code = Code(400)
desc = user=accountid_workspace_id: per-user series limit of 10000000 exceeded,
Capacity from 2,000,000 to 10,000,000 is automatically adjusted based on the last 30
 min of usage.
If throttled above 10,000,000 or in case of incoming surges, please contact
 administrator to raise it.
(local limit: 0 global limit: 10000000 actual local limit: 92879)"
```

Pour plus d'informations sur les quotas de service d'Amazon Managed Service for Prometheus et sur la manière de demander des augmentations, consultez la section Service Quotas d'Amazon Managed Service for Prometheus.

Je vois des exemples en double.

Si vous utilisez un groupe Prometheus haute disponibilité, vous devez utiliser des étiquettes externes sur vos instances Prometheus pour configurer la déduplication. Pour de plus amples informations, veuillez consulter Déduplication des métriques haute disponibilité envoyées à Amazon Managed Service for Prometheus.

D'autres problèmes liés aux données dupliquées sont abordés dans la section suivante.

Je vois des erreurs concernant les horodatages des échantillons

Amazon Managed Service for Prometheus ingère les données dans l'ordre et s'attend à ce que chaque échantillon soit horodaté plus tard que l'échantillon précédent.

Si vos données n'arrivent pas dans l'ordre, vous pouvez voir des erreurs concernant `out-of-order samples`duplicate sample for timestamp, ousamples with different value but same timestamp. Ces problèmes sont généralement dus à une configuration incorrecte du client qui envoie les données à Amazon Managed Service for Prometheus. Si vous utilisez un client Prometheus fonctionnant en mode agent, vérifiez la configuration pour détecter les règles comportant un nom de série dupliqué ou des cibles dupliquées. Si vos statistiques fournissent directement l'horodatage, vérifiez qu'elles ne sont pas hors ordre.

Vous pouvez configurer un créneau horaire pour que votre espace de travail accepte les échantillons hors service. Pour de plus amples informations, veuillez consulter Configurez votre espace de travail.

Pour plus de détails sur son fonctionnement ou sur les moyens de vérifier votre configuration, consultez le billet de blog Understanding Duplicate Samples and Out-of-order Timestamp Errors in Prometheus de Prometheus de Prom Labs.

Je vois un message d'erreur lié à une limite.

> **Note**
>
> Amazon Managed Service for Prometheus CloudWatch fournit des statistiques d'utilisation pour surveiller l'utilisation des ressources de Prometheus. À l'aide de la fonction d'alarme des métriques d' CloudWatchutilisation, vous pouvez surveiller les ressources et l'utilisation de Prometheus afin d'éviter les erreurs de limite.

Si l'un des messages d'erreur suivants s'affiche, vous pouvez demander une augmentation de l'un des quotas Amazon Managed Service for Prometheus afin de résoudre le problème. Pour de plus amples informations, veuillez consulter Service Quotas d'Amazon Managed Service for Prometheus.

- limite de série par utilisateur *<value>* dépassée, veuillez contacter l'administrateur pour l'augmenter
- limite par série métrique en cas de *<value>* dépassement, veuillez contacter l'administrateur pour l'augmenter
- limite de taux d'ingestion (...) dépassée
- la série comporte trop d'étiquettes (...) série : « % »
- la plage de temps de requête dépasse la limite (longueur de la requête : xxx, limite : yyy)
- la requête a atteint la limite maximale du nombre de morceaux lors de la récupération des morceaux auprès des ingesters
- Limite dépassée. Nombre maximal d'espaces de travail par compte.

La sortie de votre serveur Prometheus local dépasse la limite.

Amazon Managed Service for Prometheus impose des quotas de service correspondant à la quantité de données qu'un espace de travail peut recevoir des serveurs Prometheus. Pour connaître la quantité de données que votre serveur Prometheus envoie à Amazon Managed Service for Prometheus, vous pouvez exécuter les requêtes suivantes sur votre serveur Prometheus. Si vous constatez que les résultats renvoyés par Prometheus dépassent la limite fixée par Amazon Managed Service for Prometheus, vous pouvez demander une augmentation du quota de service correspondant. Pour de plus amples informations, veuillez consulter Service Quotas d'Amazon Managed Service for Prometheus.

Interrogez votre serveur Prometheus local autonome pour connaître les limites de sortie.

Type de données	Requête à utiliser	
Séries actives en cours	`prometheus_tsdb_head_series`	
Taux d'ingestion actuel	`rate(prometheus_ts`	

Type de données	Requête à utiliser	
	`db_head_samples_appended_total[5m])`	
Most-to-least liste des séries actives par nom de métrique	`sort_desc(count by(__name__)({__name__!=""}))`	
Nombre d'étiquettes par série de métriques	`group by(mylabelname)({__name__!=""})`	

Certaines de mes données n'apparaissent pas

Les données envoyées à Amazon Managed Service for Prometheus peuvent être supprimées pour diverses raisons. Le tableau suivant indique les raisons pour lesquelles les données peuvent être supprimées au lieu d'être ingérées.

Vous pouvez suivre la quantité de données supprimées et les raisons pour lesquelles elles sont supprimées à l'aide d'Amazon CloudWatch. Pour de plus amples informations, veuillez consulter Utilisez CloudWatch des métriques pour surveiller les ressources Amazon Managed Service for Prometheus.

Raison	Signification
greater_than_max_sample_age	Supprimer les lignes de journal plus anciennes que l'heure actuelle

Raison	Signification
new-value-for-timestamp	Les échantillons dupliqués sont envoyés avec le même horodatage que l'échantillon précédent, mais avec des valeurs différentes.
per_metric_series_limit	L'utilisateur a atteint la limite de séries actives par métrique.
per_user_series_limit	L'utilisateur a atteint le nombre total de séries actives.
rate_limited	Taux d'ingestion limité
sample-out-of-order	Les échantillons sont envoyés hors commande et ne peuvent pas être traités. Cette raison n'est émise que lorsque la fenêtre temporelle hors service est définie sur 0.
échantillon trop vieux	L'échantillon est plus ancien que la période de rupture configurée et ne peut pas être traité.
limite de taux de commande hors limite	Out-of-order la limite de taux d'ingestion a été atteinte et l'échantillon ne peut pas être traité. Pour de plus amples informations, veuillez consulter Service Quotas d'Amazon Managed Service for Prometheus.
label_value_too long	La valeur de l'étiquette est supérieure à la limite de caractères autorisée.
max_label_names_per_series	L'utilisateur a cliqué sur les noms d'étiquette par métrique
missing_metric_name	Le nom de la métrique n'est pas fourni.
metric_name_invalid	Nom de métrique fourni non valide.
label_invalid	Étiquette fournie non valide.
duplicate_label_names	Noms d'étiquettes fournis en double.

Raison	Signification
histogramme_native_sample_size_bytes_exceeded	L'échantillon d'histogramme natif dépasse la taille d'échantillon maximale autorisée. Réduisez le nombre de compartiments ou de travées dans l'histogramme côté client.
schéma_histogramme_natif non valide	La valeur de schéma de l'histogramme natif n'est pas valide. Les schémas valides sont compris entre -4 et 8. Vérifiez la configuration de votre client Prometheus pour connaître les paramètres. `NativeHistogramBucketFactor`
histogramme_natif_invalide	L'histogramme natif échoue à la validation (par exemple, nombre de compartiments négatif, nombre de compartiments non concordant ou travées de compartiments mal formées). Cela indique généralement un bogue dans la bibliothèque d'instrumentation.
native_histogram_buckets_exceeded	L'histogramme natif dépasse le nombre maximal de compartiments et la résolution ne peut pas être réduite automatiquement. Réduisez le nombre de buckets côté client en utilisant`NativeHistogramMaxBucketNumber` .
native_histogram_rate_limited	L'échantillon d'histogramme natif a été rejeté car la limite de taux d'ingestion de l'histogramme natif a été atteinte. Réduisez le nombre de séries d'histogrammes natifs ou la fréquence de capture, ou demandez une augmentation de la limite.
par utilisateur_native_histogram_series_limit	L'utilisateur a atteint la limite de séries actives de l'histogramme natif par espace de travail. Réduisez le nombre de séries d'histogrammes natifs ou demandez une augmentation de limite.

Erreur HTTP 422 : délai d'expiration de la requête

Si votre requête dure plus d'une minute, Amazon Managed Service for Prometheus renvoie un code d'état HTTP 422. L'exemple suivant illustre cette erreur :

```
{"status":"error","errorType":"execution","error":"query timed out - consider
 simplifying your query"}
```

Pour réduire le temps d'évaluation, appliquez une ou plusieurs des approches suivantes :

- Spécifiez une fenêtre temporelle plus courte pour évaluer moins d'échantillons.
- Ajoutez des correcteurs d'étiquettes pour réduire le nombre de séries évaluées par votre requête.
- Utilisez les règles d'enregistrement pour précalculer les agrégations coûteuses que vous demandez fréquemment.
- Évitez les correspondances de regex sur les étiquettes à forte cardinalité (étiquettes comportant de nombreuses valeurs uniques).

Pour de plus amples informations, veuillez consulter Interrogation de vos métriques.

Marquage dans Amazon Managed Service pour Prometheus

Une balise est une étiquette d'attribut personnalisée que vous attribuez ou AWS assignez à une AWS ressource. Chaque AWS étiquette comporte deux parties :

- Une clé de balise (par exemple, `CostCenter`, `Environment`, `Project` ou `Secret`). Les touches de tag distinguent les majuscules et minuscules.
- Un champ facultatif appelé valeur de balise (par exemple, `111122223333`, `Production` ou le nom d'une équipe). Omettre la valeur de balise équivaut à l'utilisation d'une chaîne vide. Les valeurs de balise sont sensibles à la casse, tout comme les clés de balise.

Ensemble, ces éléments sont connus sous le nom de paires clé-valeur. Vous pouvez attribuer jusqu'à 50 balises à chaque espace de travail.

Les balises vous aident à identifier et à organiser vos AWS ressources. De nombreux AWS services prennent en charge le balisage. Vous pouvez donc attribuer le même tag aux ressources de différents services pour indiquer que les ressources sont liées. Par exemple, vous pouvez attribuer à un espace de travail Amazon Managed Service for Prometheus la même balise qu'à un compartiment Amazon S3. Pour de plus amples informations sur le balisage des stratégies, veuillez consulter la section Balisage des ressources AWS.

Dans Amazon Managed Service for Prometheus, les espaces de travail et les espaces de noms de groupes de règles peuvent être balisés. Vous pouvez utiliser la console, le AWS CLI APIs, ou SDKs pour ajouter, gérer et supprimer des balises pour ces ressources. Outre l'identification, l'organisation et le suivi de vos espaces de travail et de vos espaces de noms de groupes de règles avec des balises, vous pouvez utiliser des balises dans les politiques IAM afin de contrôler qui peut consulter et interagir avec vos ressources Amazon Managed Service for Prometheus.

Restrictions liées aux étiquettes

Les restrictions de base suivantes s'appliquent aux balises :

- Chaque ressource peut avoir un maximum de 50 balises.
- Pour chaque ressource, chaque clé de balise doit être unique, et chaque clé de balise peut avoir une seule valeur.
- La longueur maximale des clés de balise est de 128 caractères Unicode en UTF-8.
- La longueur maximale des valeurs de balise est de 256 caractères Unicode en UTF-8.

- Si votre schéma de balisage est utilisé pour plusieurs AWS services et ressources, n'oubliez pas que d'autres services peuvent imposer des restrictions quant aux caractères autorisés. Les caractères généralement autorisés sont les lettres, les espaces et les chiffres représentables en UTF-8, ainsi que les caractères spéciaux suivants : . : + = @ _ / - (tiret).
- Les clés et les valeurs des balises distinguent les majuscules et minuscules. La bonne pratique consiste à choisir une stratégie pour mettre des balises en majuscule et mettre en œuvre cette stratégie de manière cohérente sur tous les types de ressources. Par exemple, décidez si vous souhaitez utiliser `Costcenter`, `costcenter` ou `CostCenter`, et utilisez la même convention pour toutes les balises. Évitez d'utiliser des balises avec une incohérence de traitement de cas similaires.
- N'utilisez pas `aws:`, `AWS:` ou n'importe quelle combinaison de majuscules ou minuscules de ce préfixe pour des clés ou des valeurs Ils sont réservés uniquement à AWS l'usage. Vous ne pouvez pas modifier ni supprimer des clés ou valeurs d'étiquette ayant ce préfixe. Les balises comportant ce préfixe ne sont pas prises en compte dans votre tags-per-resource limite.

Rubriques

- Tag : Amazon Managed Service pour les espaces de travail Prometheus
- Identification des espaces de noms de groupes de règles

Tag : Amazon Managed Service pour les espaces de travail Prometheus

Les balises sont des étiquettes personnalisées qui peuvent être attribuées à une ressource. Ils incluent une clé unique et une valeur facultative (dans une paire clé-valeur). Les balises vous aident à identifier et organiser vos ressources AWS . Dans Amazon Managed Service for Prometheus, les espaces de travail (et les espaces de noms des groupes de règles) peuvent être balisés. Vous pouvez utiliser la console, la AWS CLI ou SDKs pour ajouter, gérer et supprimer des balises pour ces ressources. APIs Outre l'identification, l'organisation et le suivi de vos espaces de travail à l'aide de balises, vous pouvez utiliser des balises dans les politiques IAM pour contrôler qui peut consulter et interagir avec vos ressources Amazon Managed Service for Prometheus.

Suivez les procédures de cette section pour utiliser des balises pour les espaces de travail Amazon Managed Service for Prometheus.

Rubriques

- Ajout d'une balise à un espace de travail
- Visualisation des balises d'un espace de travail
- Modification des balises d'un espace de travail
- Suppression d'une balise d'un espace de travail

Ajout d'une balise à un espace de travail

L'ajout de balises à un espace de travail Amazon Managed Service for Prometheus peut vous aider à identifier et organiser vos ressources AWS et à gérer leur accès. Tout d'abord, vous ajoutez une ou plusieurs balises (paires clé-valeur) à un espace de travail. Une fois que vous avez des balises, vous pouvez créer des politiques IAM pour gérer l'accès à l'espace de travail en fonction de ces balises. Vous pouvez utiliser la console ou le AWS CLI pour ajouter des balises à un espace de travail Amazon Managed Service for Prometheus.

> **Important**
>
> L'ajout de balises à un espace de travail peut avoir un impact sur l'accès à cet espace de travail. Avant d'ajouter une balise à un espace de travail, assurez-vous de passer en revue toutes les politiques IAM qui peuvent utiliser des balises pour contrôler l'accès aux ressources.

Pour plus d'informations sur l'ajout de balises à un espace de travail Amazon Managed Service for Prometheus lorsque vous le créez, consultez la section Création d'un service géré Amazon pour l'espace de travail Prometheus.

Rubriques

- Ajout d'une balise à un espace de travail (console)
- Ajout d'une balise à un espace de travail (AWS CLI)

Ajout d'une balise à un espace de travail (console)

Vous pouvez utiliser la console pour ajouter une ou plusieurs balises à un espace de travail Amazon Managed Service for Prometheus.

1. Ouvrez la console Amazon Managed Service for Prometheus à l'adresse. https://console.aws.amazon.com/prometheus/

2. Dans le volet de navigation, choisissez l'icône de menu.
3. Sélectionnez All workspaces.
4. Choisissez l'ID de l'espace de travail que vous voulez gérer.
5. Sélectionnez l'onglet Tags (Identifications).
6. Si aucune balise n'a été ajoutée à l'espace de travail Amazon Managed Service for Prometheus, sélectionnez Create tag. Sinon, sélectionnez Gérer les balises.
7. Dans Key (Clé), entrez un nom de balise. Vous pouvez ajouter une valeur en option pour la balise dans Value (Valeur).
8. (Facultatif) Pour ajouter une autre balise, choisissez à nouveau Add tag (Ajouter une balise).
9. Une fois les balises ajoutées, choisissez Enregistrer les modifications.

Ajout d'une balise à un espace de travail (AWS CLI)

Suivez ces étapes pour utiliser le AWS CLI pour ajouter une balise à un espace de travail Amazon Managed Service for Prometheus. Pour ajouter une balise à un espace de travail lors de sa création, veuillez consulter la section Création d'un service géré Amazon pour l'espace de travail Prometheus.

Au cours de ces étapes, nous partons du principe que vous avez déjà installé une version récente AWS CLI ou que vous avez déjà mis à jour la version actuelle. Pour plus d'informations, consultez Installing the AWS Command Line Interface (Installation de).

Depuis le terminal ou la ligne de commande, exécutez la commande tag-resource, en spécifiant l'ARN (Amazon Resource Name) de l'espace de travail dans lequel vous souhaitez ajouter des balises ainsi que la clé et la valeur de la balise que vous souhaitez ajouter. Vous pouvez ajouter plusieurs balises à un espace de travail. Par exemple, pour étiqueter un espace de travail Amazon Managed Service for Prometheus nommé My-Workspace avec deux balises, une clé de balise *Status* nommée avec la valeur *Secret* de balise et une clé de balise *Team* nommée avec la valeur de balise de : *My-Team*

```
aws amp tag-resource --resource-arn arn:aws:aps:us-
west-2:123456789012:workspaces/IDstring
--tags Status=Secret,Team=My-Team
```

Si elle aboutit, cette commande ne renvoie rien.

Visualisation des balises d'un espace de travail

Les balises peuvent vous aider à identifier et à organiser vos AWS ressources, ainsi qu'à gérer l'accès à celles-ci. Pour plus d'informations sur les stratégies de balisage, consultez la section Ressources de balisage. AWS

Visualisation des balises pour un espace de travail Amazon Managed Service for Prometheus (console)

Vous pouvez utiliser la console pour afficher les balises associées à un espace de travail Amazon Managed Service for Prometheus.

1. Ouvrez la console Amazon Managed Service for Prometheus à l'adresse. https://console.aws.amazon.com/prometheus/
2. Dans le volet de navigation, choisissez l'icône de menu.
3. Sélectionnez All workspaces.
4. Choisissez l'ID de l'espace de travail que vous voulez gérer.
5. Sélectionnez l'onglet Tags (Identifications).

Visualisation des balises pour un espace de travail Amazon Managed Service for Prometheus (AWS CLI)

Procédez comme suit pour utiliser le AWS CLI pour afficher les AWS balises d'un espace de travail. Si aucune balise n'a été ajoutée, la liste renvoyée est vide.

Depuis le terminal ou la ligne de commande, exécutez la commande list-tags-for-resource. Par exemple, pour afficher la liste des clés et des valeurs de balise pour un espace de travail :

```
aws amp list-tags-for-resource --resource-arn arn:aws:aps:us-
west-2:123456789012:workspace/IDstring
```

Si elle aboutit, cette commande renvoie des informations similaires à ce qui suit :

```
{
    "tags": {
        "Status": "Secret",
        "Team": "My-Team"
```

```
    }
}
```

Modification des balises d'un espace de travail

Vous pouvez modifier la valeur d'une balise associée à un espace de travail. Vous pouvez également modifier le nom de la clé, ce qui équivaut à supprimer la balise et à ajoutant une carte différente avec le nouveau nom et la même valeur que l'autre clé.

> **Important**
>
> La modification des balises d'un espace de travail Amazon Managed Service for Prometheus peut avoir un impact sur cet espace de travail. Avant de modifier le nom (clé) ou la valeur d'une balise pour un espace de travail, assurez-vous de passer en revue toutes les politiques IAM qui peuvent utiliser la clé ou la valeur d'une balise pour contrôler l'accès aux ressources, telles que les référentiels.

Modification d'une balise pour un espace de travail Amazon Managed Service for Prometheus (console)

Vous pouvez utiliser la console pour modifier les balises associées à un espace de travail Amazon Managed Service for Prometheus.

1. Ouvrez la console Amazon Managed Service for Prometheus à l'adresse. https://console.aws.amazon.com/prometheus/
2. Dans le volet de navigation, choisissez l'icône de menu.
3. Sélectionnez All workspaces.
4. Choisissez l'ID de l'espace de travail que vous voulez gérer.
5. Sélectionnez l'onglet Tags (Identifications).
6. Si aucune balise n'a été ajoutée à l'espace de travail, sélectionnez Create tag. Sinon, sélectionnez Gérer les balises.
7. Dans Key (Clé), entrez un nom de balise. Vous pouvez ajouter une valeur en option pour la balise dans Value (Valeur).
8. (Facultatif) Pour ajouter une autre balise, choisissez à nouveau Add tag (Ajouter une balise).

9. Une fois les balises ajoutées, choisissez Enregistrer les modifications.

Modification des balises pour un espace de travail Amazon Managed Service for Prometheus (AWS CLI)

Procédez comme suit pour utiliser le AWS CLI pour mettre à jour une balise pour un espace de travail. Vous pouvez modifier la valeur d'une clé existante ou ajouter une autre clé.

Depuis le terminal ou la ligne de commande, exécutez la commande tag-resource, en spécifiant l'ARN (Amazon Resource Name) de l'espace de travail Amazon Managed Service for Prometheus dans lequel vous souhaitez mettre à jour une balise et spécifiez la clé de balise et la valeur de balise :

```
aws amp tag-resource --resource-arn arn:aws:aps:us-
west-2:123456789012:workspace/IDstring --tags Team=New-Team
```

Suppression d'une balise d'un espace de travail

Vous pouvez supprimer une ou plusieurs balises associées à un espace de travail. La suppression d'un tag ne supprime pas le tag des autres AWS ressources associées à ce tag.

> **Important**
>
> La suppression des balises d'un espace de travail Amazon Managed Service for Prometheus peut avoir un impact sur cet espace de travail. Avant de supprimer une balise d'un espace de travail, assurez-vous de passer en revue toutes les politiques IAM qui peuvent utiliser la clé ou la valeur d'une balise pour contrôler l'accès aux ressources, telles que les référentiels.

Suppression d'une balise d'un espace de travail Amazon Managed Service for Prometheus (console)

Vous pouvez utiliser la console pour supprimer l'association entre une balise et un espace de travail.

1. Ouvrez la console Amazon Managed Service for Prometheus à l'adresse. https://console.aws.amazon.com/prometheus/
2. Dans le volet de navigation, choisissez l'icône de menu.
3. Sélectionnez All workspaces.

4. Choisissez l'ID de l'espace de travail que vous voulez gérer.
5. Sélectionnez l'onglet Tags (Identifications).
6. Choisissez Gérer les balises.
7. Recherchez la balise que vous souhaitez supprimer, puis sélectionnez Supprimer.

Suppression d'une balise d'un espace de travail Amazon Managed Service for Prometheus (AWS CLI)

Procédez comme suit pour utiliser le AWS CLI pour supprimer une étiquette d'un espace de travail. La suppression d'une balise supprime uniquement son association à l'espace de travail, mais pas la balise en elle-même.

> **Note**
>
> Si vous supprimez un espace de travail Amazon Managed Service for Prometheus, toutes les associations de balises sont supprimées de l'espace de travail supprimé. Vous n'avez pas besoin de supprimer les balises avant de supprimer un espace de travail.

Depuis le terminal ou la ligne de commande, exécutez la commande untag-resource, en spécifiant l'ARN (Amazon Resource Name) de l'espace de travail dans lequel vous souhaitez supprimer des balises et la clé de la balise que vous souhaitez supprimer. Par exemple, pour supprimer une balise sur un espace de travail nommé My-Workspace à l'aide de la clé *`Status`* de balise :

```
aws amp untag-resource --resource-arn arn:aws:aps:us-
west-2:123456789012:workspace/IDstring --tag-keys Status
```

Si elle aboutit, cette commande ne renvoie rien. Pour vérifier quelles balises sont associées à l'espace de travail, exécutez la commande list-tags-for-resource.

Identification des espaces de noms de groupes de règles

Les balises sont des étiquettes personnalisées qui peuvent être attribuées à une ressource. Ils incluent une clé unique et une valeur facultative (dans une paire clé-valeur). Les balises vous aident à identifier et organiser vos ressources AWS . Dans Amazon Managed Service for Prometheus, les espaces de noms (et espaces de travail) des groupes de règles peuvent être balisés. Vous

pouvez utiliser la console, la AWS CLI ou SDKs pour ajouter, gérer et supprimer des balises pour ces ressources. APIs Outre l'identification, l'organisation et le suivi des espaces de noms de vos groupes de règles à l'aide de balises, vous pouvez utiliser des balises dans les politiques IAM pour contrôler les personnes autorisées à consulter et à interagir avec vos ressources Amazon Managed Service for Prometheus.

Suivez les procédures de cette section pour utiliser des balises pour les espaces de noms de groupes de règles Amazon Managed Service for Prometheus.

Rubriques

- Ajouter une balise à un espace de noms de groupes de règles
- Visualisation des balises d'un espace de noms de groupes de règles
- Modification des balises d'un espace de noms de groupes de règles
- Suppression d'une balise d'un espace de noms de groupes de règles

Ajouter une balise à un espace de noms de groupes de règles

L'ajout de balises aux espaces de noms des groupes de règles Amazon Managed Service for Prometheus peut vous aider à identifier et à organiser AWS vos ressources, ainsi qu'à gérer l'accès à celles-ci. Tout d'abord, vous ajoutez une ou plusieurs balises (paires clé-valeur) à un espace de noms de groupes de règles. Une fois que vous avez des balises, vous pouvez créer des politiques IAM pour gérer l'accès à l'espace de noms en fonction de ces balises. Vous pouvez utiliser la console ou le AWS CLI pour ajouter des balises à un espace de noms de groupes de règles Amazon Managed Service for Prometheus.

> **Important**
>
> L'ajout de balises à un espace de noms de groupes de règles peut avoir un impact sur l'accès à cet espace de noms de groupes de règles. Avant d'ajouter une balise, assurez-vous de passer en revue toutes les politiques IAM qui peuvent utiliser des balises pour contrôler l'accès aux ressources.

Pour plus d'informations sur l'ajout de balises à un espace de noms de groupes de règles lorsque vous le créez, consultez Création d'un fichier de règles.

Rubriques

- Ajouter une balise à un espace de noms de groupes de règles (console)
- Ajouter une balise à l'espace de noms de groupes de règles (AWS CLI)

Ajouter une balise à un espace de noms de groupes de règles (console)

Vous pouvez utiliser la console pour ajouter une ou plusieurs balises à un espace de noms de groupes de règles Amazon Managed Service for Prometheus.

1. Ouvrez la console Amazon Managed Service for Prometheus à l'adresse. https://console.aws.amazon.com/prometheus/
2. Dans le volet de navigation, choisissez l'icône de menu.
3. Sélectionnez All workspaces.
4. Choisissez l'ID de l'espace de travail que vous voulez gérer.
5. Choisissez l'onglet Rules management.
6. Cliquez sur le bouton en regard du nom de l'espace de noms, puis sélectionnez Modifier.
7. Sélectionnez Create tags, Ajouter une nouvelle balise.
8. Dans Key (Clé), entrez un nom de balise. Vous pouvez ajouter une valeur en option pour la balise dans Value (Valeur).
9. (Facultatif) Pour ajouter une autre balise, sélectionnez à nouveau Ajouter une nouvelle balise.
10. Une fois les balises ajoutées, choisissez Enregistrer les modifications.

Ajouter une balise à l'espace de noms de groupes de règles (AWS CLI)

Suivez ces étapes pour utiliser le pour ajouter une balise AWS CLI à un espace de noms de groupes de règles Amazon Managed Service for Prometheus. Pour ajouter une balise à un espace de noms de groupes de règles lorsque vous le créez, consultez la section Chargez un fichier de configuration des règles sur Amazon Managed Service for Prometheus.

Au cours de ces étapes, nous partons du principe que vous avez déjà installé une version récente AWS CLI ou que vous avez déjà mis à jour la version actuelle. Pour plus d'informations, consultez Installing the AWS Command Line Interface (Installation de).

Depuis le terminal ou la ligne de commande, exécutez la commande tag-resource, en spécifiant l'ARN (Amazon Resource Name) de l'espace de noms de groupes de règles dans lequel vous souhaitez ajouter des balises ainsi que la clé et la valeur de la balise que vous souhaitez ajouter.

Vous pouvez ajouter plusieurs balises à un espace de noms de groupes de règles. Par exemple, pour baliser un espace de noms Amazon Managed Service for Prometheus nommé My-Workspace avec deux balises, une clé de balise *Status* nommée avec la valeur de balise et une clé *Secret* de balise *Team* nommée avec la valeur de balise de : *My-Team*

```
aws amp tag-resource \
    --resource-arn arn:aws:aps:us-
west-2:123456789012:rulegroupsnamespace/IDstring/namespace_name \
    --tags Status=Secret,Team=My-Team
```

Si elle aboutit, cette commande ne renvoie rien.

Visualisation des balises d'un espace de noms de groupes de règles

Les balises peuvent vous aider à identifier et à organiser vos AWS ressources, ainsi qu'à gérer l'accès à celles-ci. Pour plus d'informations sur les stratégies de balisage, consultez la section Ressources de balisage. AWS

Visualisation des balises d'un espace de noms de groupes de règles Amazon Managed Service for Prometheus (console)

Vous pouvez utiliser la console pour afficher les balises associées à un espace de noms de groupes de règles Amazon Managed Service for Prometheus.

1. Ouvrez la console Amazon Managed Service for Prometheus à l'adresse. https://console.aws.amazon.com/prometheus/
2. Dans le volet de navigation, choisissez l'icône de menu.
3. Sélectionnez All workspaces.
4. Choisissez l'ID de l'espace de travail que vous voulez gérer.
5. Choisissez l'onglet Rules management.
6. Sélectionnez le nom de l'espace de noms.

Visualisation des balises pour un espace de travail Amazon Managed Service for Prometheus (AWS CLI)

Procédez comme suit pour utiliser le AWS CLI pour afficher les AWS balises d'un espace de noms de groupes de règles. Si aucune balise n'a été ajoutée, la liste renvoyée est vide.

Depuis le terminal ou la ligne de commande, exécutez la commande list-tags-for-resource. Par exemple, pour afficher la liste des clés et des valeurs de balise pour un espace de noms de groupes de règles :

```
aws amp list-tags-for-resource --resource-arn rn:aws:aps:us-
west-2:123456789012:rulegroupsnamespace/IDstring/namespace_name
```

Si elle aboutit, cette commande renvoie des informations similaires à ce qui suit :

```
{
    "tags": {
        "Status": "Secret",
        "Team": "My-Team"
    }
}
```

Modification des balises d'un espace de noms de groupes de règles

Vous pouvez modifier la valeur d'une balise associée à un espace de noms de groupes de règles. Vous pouvez également modifier le nom de la clé, ce qui équivaut à supprimer la balise et à ajoutant une carte différente avec le nouveau nom et la même valeur que l'autre clé.

> **Important**
>
> La modification des balises d'un espace de noms de groupes de règles peut avoir un impact sur l'accès à ce dernier. Avant de modifier le nom (clé) ou la valeur d'une balise pour une ressource, assurez-vous de passer en revue toutes les politiques IAM qui peuvent utiliser la clé ou la valeur d'une balise pour contrôler l'accès aux ressources.

Modification d'une balise d'un espace de noms de groupes de règles Amazon Managed Service for Prometheus (console)

Vous pouvez utiliser la console pour modifier les balises associées à un espace de noms de groupes de règles Amazon Managed Service for Prometheus.

1. Ouvrez la console Amazon Managed Service for Prometheus à l'adresse. https://console.aws.amazon.com/prometheus/

2. Dans le volet de navigation, choisissez l'icône de menu.
3. Sélectionnez All workspaces.
4. Choisissez l'ID de l'espace de travail que vous voulez gérer.
5. Choisissez l'onglet Rules management.
6. Choisissez le nom de l'espace de noms.
7. Choisissez Gérer les balises et Ajouter une nouvelle balise.
8. Pour modifier la valeur d'une balise existante, saisissez la nouvelle valeur dans Valeur.
9. Pour ajouter une balise supplémentaire, choisissez Ajouter une nouvelle balise.
10. Une fois les balises ajoutées et modifiées, choisissez Enregistrer les modifications.

Modification des balises d'un espace de noms de groupes de règles Amazon Managed Service for Prometheus (AWS CLI)

Procédez comme suit pour utiliser le AWS CLI pour mettre à jour une balise pour un espace de noms de groupes de règles. Vous pouvez modifier la valeur d'une clé existante ou ajouter une autre clé.

Depuis le terminal ou la ligne de commande, exécutez la commande tag-resource, en spécifiant l'ARN (Amazon Resource Name) de la ressource dans laquelle vous souhaitez mettre à jour une balise et spécifiez la clé de balise et la valeur de balise :

```
aws amp tag-resource --resource-arn rn:aws:aps:us-
west-2:123456789012:rulegroupsnamespace/IDstring/namespace_name --tags Team=New-Team
```

Suppression d'une balise d'un espace de noms de groupes de règles

Vous pouvez supprimer une ou plusieurs balises associées à un espace de noms de groupes de règles. La suppression d'un tag ne supprime pas le tag des autres AWS ressources associées à ce tag.

> **Important**
>
> La suppression de balises associées à une ressource peut avoir un impact sur l'accès à cette ressource. Avant de supprimer une balise d'une ressource, assurez-vous de passer en revue toutes les politiques IAM qui peuvent utiliser la clé ou la valeur d'une balise pour contrôler l'accès aux ressources, telles que les référentiels.

Suppression d'une balise d'un espace de noms de groupes de règles Amazon Managed Service for Prometheus (console)

Vous pouvez utiliser la console pour supprimer l'association entre une balise et un espace de noms de groupes de règles.

1. Ouvrez la console Amazon Managed Service for Prometheus à l'adresse. https://console.aws.amazon.com/prometheus/
2. Dans le volet de navigation, choisissez l'icône de menu.
3. Sélectionnez All workspaces.
4. Choisissez l'ID de l'espace de travail que vous voulez gérer.
5. Choisissez l'onglet Rules management.
6. Choisissez le nom de l'espace de noms.
7. Choisissez Gérer les balises.
8. En regard de la balise que vous souhaitez supprimer, sélectionnez Supprimer.
9. Lorsque vous avez terminé, choisissez Enregistrer les modifications.

Suppression d'une balise d'un espace de noms de groupes de règles Amazon Managed Service for Prometheus (AWS CLI)

Procédez comme suit pour utiliser le AWS CLI pour supprimer une balise de l'espace de noms d'un groupe de règles. La suppression d'une balise supprime uniquement son association à l'espace de noms de groupes de règles, mais pas la balise en elle-même.

> **Note**
>
> Si vous supprimez un espace de noms de groupes de règles Amazon Managed Service for Prometheus, toutes les associations de balises sont supprimées de l'espace de noms supprimé. Vous n'avez pas besoin de supprimer les balises avant de supprimer un espace de noms.

Depuis le terminal ou la ligne de commande, exécutez la commande untag-resource, en spécifiant l'ARN (Amazon Resource Name) de l'espace de noms de groupes de règles dans lequel vous souhaitez supprimer des balises et la clé de la balise que vous souhaitez supprimer. Par exemple,

pour supprimer une balise sur un espace de travail nommé My-Workspace à l'aide de la clé *Status* de balise :

```
aws amp untag-resource --resource-arn rn:aws:aps:us-
west-2:123456789012:rulegroupsnamespace/IDstring/namespace_name --tag-keys Status
```

Si elle aboutit, cette commande ne renvoie rien. Pour vérifier les balises associées à la ressource, exécutez la commande list-tags-for-resource.

Service Quotas d'Amazon Managed Service for Prometheus

Les deux sections suivantes décrivent les quotas et limites associés à Amazon Managed Service for Prometheus.

Quotas de service

Amazon Managed Service for Prometheus comporte les quotas suivants. Amazon Managed Service for Prometheus fournit des statistiques d'utilisation pour surveiller l'CloudWatch utilisation des ressources de Prometheus. À l'aide de la fonction d'alarme des métriques d' CloudWatch utilisation d'Amazon, vous pouvez surveiller les ressources et l'utilisation de Prometheus afin d'éviter les erreurs de limite.

Au fur et à mesure que vos projets et espaces de travail se développent, les quotas les plus courants que vous devez surveiller ou pour lesquels vous devez demander une augmentation sont les suivants : séries actives par espace de travail et taux d'ingestion par espace de travail.

Pour tous les quotas ajustables, vous pouvez demander une augmentation de quota en cliquant sur le lien dans la colonne Réglable ou en demandant une augmentation de quota.

La limite de séries actives par espace de travail s'applique dynamiquement. Pour plus d'informations, consultez Quotas par défaut des séries actives. Le taux d'ingestion par quota d'espace de travail détermine la rapidité avec laquelle vous pouvez ingérer des données dans votre espace de travail. Pour plus d'informations, consultez Régulation de l'ingestion.

> **Note**
>
> Sauf indication contraire, ces quotas s'entendent par espace de travail. La valeur maximale des séries actives par espace de travail est de 1,5 milliard.

Nom	Par défaut	Ajusta	Description
Métriques actives avec métadonnées par espace de travail	Chaque Région prise en charge : 20 000	Non	Nombre de métriques actives uniques avec métadonnées par espace de travail. Remarque :

Nom	Par défaut	Ajusta	Description
			Si la limite est atteinte, l'échantillon métrique est enregistré, mais les métadonnées dépassant cette limite sont supprimées.
Séries actives par espace de travail	Chaque Région prise en charge : 50 000 000	Oui	Le nombre de séries actives uniques par espace de travail (jusqu'à un maximum de 1,5 milliard). Une série est active si un échantillon a été signalé au cours des 2 dernières heures. La capacité de 2 M à 50 M est automatiquement ajustée en fonction des 30 dernières minutes d'utilisation.
Taille du groupe d'agrégation d'alertes dans le fichier de définition du gestionnaire d'alertes	Chaque Région prise en charge : 1 000	Oui	Taille maximale d'un groupe d'agrégation d'alertes dans le fichier de définition du gestionnaire d'alertes. Chaque combinaison de valeurs d'étiquette group_by crée un groupe d'agrégation.
Taille du fichier de définition du gestionnaire d'alertes	Chaque région prise en charge : 1 000 000	Non	Taille maximale d'un fichier de définition du gestionnaire d'alertes, en octets.

Nom	Par défaut	Ajusta	Description
Taille de la charge utile des alertes dans Alert Manager	Chaque région prise en charge : 20 000 000	Non	Taille maximale de la charge utile de toutes les alertes Alert Manager par espace de travail, en octets. La taille de l'alerte dépend des étiquettes et des annotations.
Alertes dans le gestionnaire d'alertes	Chaque Région prise en charge : 1 000	Oui	Nombre maximal d'alertes Alert Manager simultanées par espace de travail.
Clusters de suivi de la haute disponibi lité	Chaque région prise en charge : 500	Non	Nombre maximal de clusters que le dispositi f de suivi de la haute disponibilité suivra pour les échantillons ingérés par espace de travail.
Taux d'ingestion par espace de travail	Chaque région prise en charge : 1 666 666	Oui	Taux d'ingestion d'échanti llons de métriques par espace de travail et par seconde. La limite est automatiquement ajustée pour correspondre à 1/30 la série active par limite d'espace de travail, jusqu'à 1 666 666.
Règles d'inhibition dans le fichier de définition du gestionnaire d'alertes	Chaque région prise en charge : 100	Oui	Nombre maximal de règles d'inhibition dans le fichier de définition du gestionnaire d'alertes.

Nom	Par défaut	Ajusta	Description
Taille de l'étiquette	Chaque région prise en charge : 7	Non	Taille combinée maximale de toutes les étiquette s et valeurs d'étiquette acceptées pour une série, en kilo-octets.
LabelSet limites par espace de travail	Chaque région prise en charge : 100	Oui	Nombre maximum de limites de jeux d'étiquet tes pouvant être créées par espace de travail.
Étiquettes par série de métriques	Chaque Région prise en charge : 150	Oui	Nombre d'étiquettes par série de métriques.
Nombre maximal de compartiments pour l'histogramme natif	Chaque région prise en charge : 200	Non	Le nombre total maximum de compartiments autorisés dans un échantillon d'histogr amme natif. La résolutio n d'un histogramme comportant plusieurs compartiments sera réduite jusqu'à ce que le nombre de compartim ents atteigne la limite.
Longueur des métadonnées	Par région prise en charge : 1	Non	Longueur maximale acceptée pour les métadonnées métriques , en kilo-octets. Les métadonnées font référence au nom, au type, à l'unité et au texte d'aide de la métrique.

Nom	Par défaut	Ajusta	Description
Métadonnées par métrique	Chaque Région prise en charge : 10	Non	Nombre de métadonnées par métrique. Remarque : Si la limite est atteinte, l'échantillon métrique est enregistré, mais les métadonnées dépassant cette limite sont supprimées.
Histogramme natif : séries actives par espace de travail	Chaque région prise en charge : 5 000 000	Oui	Le nombre de séries actives uniques de Native Histogram par espace de travail. Une série est active si un échantillon a été signalé au cours des 2 dernières heures. La limite est automatiquement ajustée à 10 % de la limite de série active des espaces de travail, ou à la valeur par défaut, selon la valeur la plus élevée.
Taux d'ingestion de l'histogramme natif par espace de travail	Chaque région prise en charge : 166 666	Oui	Taux d'ingestion d'échantillons de l'histogramme natif par espace de travail et par seconde. La limite est automatiquement ajustée à 10 % de la limite du taux d'ingestion des espaces de travail, ou à la valeur par défaut, la valeur la plus élevée étant retenue.

Nom	Par défaut	Ajusta	Description
Nœuds dans l'arborescence de routage du gestionnaire d'alertes	Chaque région prise en charge : 100	Oui	Nombre maximal de nœuds dans l'arbores cence de routage du gestionnaire d'alertes.
Nombre d'opérations d'API par région en transactions par seconde	Par région prise en charge : 10	Oui	Le nombre maximum d'opérations d'API par seconde et par région pour toutes les API Amazon Managed Service for Prometheus, y compris les API CRUD d'espace de travail, les API de balisage, les API CRUD d'espace de noms de groupes de règles et les API CRUD de définition du gestionnaire d'alertes.
Nombre d' GetSeriesopérations GetLabels et d' GetMetricMetadata API par espace de travail en transactions par seconde	Chaque Région prise en charge : 10	Non	Le nombre maximum d' GetSeriesopérations GetLabels et d' GetMetric Metadata Prometheu s-compatible API par seconde et par espace de travail.
Nombre d'opérations d' QueryMetrics API par espace de travail en transacti ons par seconde	Chaque région prise en charge : 300	Oui	Le nombre maximum d'opérations d' QueryMetr ics Prometheus-compati ble API par seconde et par espace de travail.

Nom	Par défaut	Ajusta	Description
Nombre d'opérations d' RemoteWrite API par espace de travail en transactions par seconde	Chaque région prise en charge : 3 000	Oui	Le nombre maximum d'opérations d' RemoteWrite Prometheus-compatible API par seconde et par espace de travail.
Nombre d'autres opérations d' Prometheus-compatible API par espace de travail en transactions par seconde	Chaque région prise en charge : 100	Non	Le nombre maximum d'opérations d'API par seconde et par espace de travail pour toutes les autres Prometheus-compatible API ListAlerts ListRules, y compris, etc.
Taux d'ingestion de produits hors service par espace de travail	Chaque région prise en charge : 83 333	Oui	Taux d'ingestion d'échantillons hors service par espace de travail et par seconde. À moins d'être remplacée, la limite est automatiquement ajustée pour correspondre à 5 % du taux d'ingestion par limite d'espace de travail.

Nom	Par défaut	Ajusta	Description
Fenêtre de temps hors service par espace de travail	Chaque région prise en charge : 600	Oui	Période maximale pour les échantillons hors service par espace de travail, en secondes. La fenêtre de temps hors service par défaut lorsqu'un nouvel espace de travail est créé est de 60 secondes, et elle peut être configurée jusqu'à 600 secondes au maximum.
Octets de requête pour les requêtes instantanées	Chaque région prise en charge : 5	Non	Nombre maximal d'octets pouvant être analysés par une seule requête instantanée, en gigaoctets.
Octets de requête pour les requêtes de plage	Chaque région prise en charge : 5	Non	Nombre maximal d'octets pouvant être analysés par intervalle de 24 heures dans le cadre d'une requête à plage unique, en gigaoctets.
Exemples de requête	Chaque Région prise en charge : 50 000 000	Non	Nombre maximal d'échantillons pouvant être numérisés par intervalle de 24 heures dans le cadre d'une seule requête par plage ou d'une seule requête instantanée.

Nom	Par défaut	Ajusta	Description
Série de requêtes récupérée	Chaque Région prise en charge : 12 000 000	Non	Nombre maximal de séries pouvant être numérisées par intervalle de 24 heures dans le cadre d'une seule requête par plage ou d'une seule requête instantanée.
Plage de temps de requête en jours	Chaque région prise en charge : 95	Non	La plage de temps maximale de QueryMetrics GetSeries, et les GetLabels API.
Taille des demandes	Par région prise en charge : 1	Non	Taille maximale de requête pour l'ingestion ou la requête, en mégaoctets.
Intervalle d'évaluation des règles	Chaque Région prise en charge : 30	Oui	Intervalle minimal d'évaluation des règles d'un groupe de règles par espace de travail, en secondes.
Taille du fichier de définition de l'espace de noms de groupes de règles	Chaque région prise en charge : 1 000 000	Non	Taille maximale d'un fichier de définition d'espace de noms de groupe de règles, en octets.
Règles par espace de travail	Chaque Région prise en charge : 2 000	Oui	Nombre maximal de règles par espace de travail.

Nom	Par défaut	Ajusta	Description
Silences par espace de travail	Chaque Région prise en charge : 1 000	Oui	Nombre maximum de silences, y compris les silences expirés, actifs et en attente, par espace de travail.
Modèles dans le fichier de définition du gestionnaire d'alertes	Chaque région prise en charge : 100	Oui	Nombre maximal de modèles dans le fichier de définition du gestionnaire d'alertes.
Espaces de travail par région et par compte	Chaque région prise en charge : 25	Oui	Nombre maximal d'espaces de travail par région.

Quotas par défaut des séries actives

Les espaces de travail Amazon Managed Service for Prometheus s'adaptent automatiquement à votre consommation d'ingestion. À mesure que votre utilisation augmente, le service augmente automatiquement la capacité de vos séries chronologiques jusqu'au quota par défaut.

Votre espace de travail Amazon Managed Service for Prometheus évolue automatiquement, en fonction de votre utilisation, de deux manières :

1. Lorsque votre utilisation moyenne sur 30 minutes est inférieure à 5 millions de séries, la capacité double (par exemple, un espace de travail dont l'utilisation est de 3,5 millions atteint une capacité de 7 millions).
2. Lorsque l'utilisation dépasse 5 millions de séries, l'espace de travail ajoute une mémoire tampon de 10 millions (par exemple, un espace de travail dont l'utilisation est de 25 millions obtient une capacité de 35 millions).

Amazon Managed Service for Prometheus alloue automatiquement une plus grande capacité à mesure que votre ingestion augmente, dans la limite de votre quota. Cela permet de garantir que votre charge de travail ne soit pas ralentie de manière prolongée. Toutefois, un ralentissement peut se produire si vous doublez ou dépassez 10 millions de dollars par rapport à la valeur de référence

précédente calculée au cours des 30 dernières minutes. Pour éviter tout ralentissement, Amazon Managed Service for Prometheus recommande d'augmenter progressivement l'ingestion lorsque vous dépassez votre niveau de référence précédent.

> **Note**
>
> La capacité minimale pour les séries chronologiques actives est de 2 millions, et il n'y a pas de limitation lorsque vous en avez moins de 2 millions.
> Pour dépasser votre quota par défaut, vous pouvez demander une augmentation de quota.

Dépasser le quota par défaut

Lorsque vous demandez une augmentation de quota supérieure au quota de séries actives par défaut, Amazon Managed Service for Prometheus ajuste la capacité de votre espace de travail en conséquence. Si vous n'utilisez pas pleinement la capacité accrue, le service récupérera la partie inutilisée au fil du temps. Au fur et à mesure que votre utilisation augmente, l'espace de travail s'agrandira automatiquement.

Toutefois, la régulation peut se produire si vous doublez ou dépassez 50 millions de séries chronologiques actives par rapport à votre base de référence précédente calculée au cours des 2 dernières heures. Par exemple :

- Si votre quota est de 100 millions et que votre base de référence est de 30 millions, vous pouvez passer à 60 millions en 2 heures sans limitation.
- Si votre quota est de 100 millions et que votre base de référence est de 50 millions, vous pouvez passer aux 100 millions en 2 heures sans limitation.

Régulation de l'ingestion

Amazon Managed Service for Prometheus limite l'ingestion pour chaque espace de travail, en fonction de vos limites actuelles. Cela permet de maintenir les performances de l'espace de travail. Si vous dépassez la limite, vous le verrez `DiscardedSamples` dans CloudWatch les statistiques (avec la `rate_limited` raison). Vous pouvez l'utiliser CloudWatch pour surveiller votre ingestion et créer une alarme pour vous avertir lorsque vous êtes sur le point d'atteindre les limites d'étranglement. Pour plus d'informations, consultez Utilisez CloudWatch des métriques pour surveiller les ressources Amazon Managed Service for Prometheus.

Amazon Managed Service for Prometheus utilise l'algorithme Token Bucket pour implémenter la régulation de l'ingestion. Avec cet algorithme, votre compte dispose d'un compartiment contenant un nombre spécifique de jetons. Le nombre de jetons contenus dans le bucket représente votre limite d'ingestion à chaque seconde.

Chaque échantillon de données ingéré supprime un jeton du compartiment. Si la taille de votre bucket (taux d'ingestion par espace de travail) est de 1 000 000, votre espace de travail peut ingérer un million d'échantillons de données en une seconde. S'il dépasse un million d'échantillons à ingérer, il sera limité et aucun autre enregistrement ne sera ingéré. Les échantillons de données supplémentaires seront supprimés.

Le seau se recharge automatiquement à un débit défini. Si le compartiment est inférieur à sa capacité maximale, un nombre défini de jetons y est ajouté chaque seconde jusqu'à ce qu'il atteigne sa capacité maximale. Si le seau est plein à l'arrivée des jetons de recharge, ils sont jetés. Le bucket ne peut pas contenir plus de jetons que son maximum. Le taux de recharge pour l'ingestion des échantillons est défini par la limite du taux d'ingestion par espace de travail. Si votre taux d'ingestion par espace de travail est fixé à 170 000, le taux de recharge du bucket est de 170 000 jetons par seconde.

Si votre espace de travail ingère 1 000 000 d'échantillons de données par seconde, votre bucket est immédiatement réduit à zéro jeton. Le seau est ensuite rempli de 170 000 jetons par seconde, jusqu'à ce qu'il atteigne sa capacité maximale de 1 000 000 de jetons. S'il n'y a plus d'ingestion, le seau précédemment vide retrouvera sa capacité maximale en 6 secondes.

> **Note**
>
> L'ingestion se produit dans le cadre de demandes groupées. Si vous avez 100 jetons disponibles et que vous envoyez une demande contenant 101 échantillons, l'intégralité de la demande est rejetée. Amazon Managed Service for Prometheus n'accepte pas partiellement les demandes. Si vous rédigez un collecteur, vous pouvez gérer les nouvelles tentatives (avec des lots plus petits ou après un certain temps écoulé).

Il n'est pas nécessaire d'attendre que le compartiment soit plein pour que votre espace de travail puisse ingérer d'autres échantillons de données. Vous pouvez utiliser des jetons au fur et à mesure qu'ils sont ajoutés au bucket. Si vous utilisez immédiatement les jetons de recharge, le seau n'atteint pas sa capacité maximale. Par exemple, si vous épuisez le compartiment, vous pouvez continuer

à ingérer 170 000 échantillons de données par seconde. Le seau ne peut être rempli à sa capacité maximale que si vous ingérez moins de 170 000 échantillons de données par seconde.

Limites supplémentaires relatives aux données ingérées

Amazon Managed Service for Prometheus impose également les exigences supplémentaires suivantes pour les données ingérées dans l’espace de travail. Ces exigences ne sont pas ajustables.

- L’ingestion d’échantillons de métriques datant de plus d’une heure est refusée.
- Chaque échantillon et chaque métadonnée doivent avoir un nom de métrique.
- Les échantillons d'histogrammes natifs doivent avoir une valeur de schéma comprise entre -4 et 8. Les échantillons dont le schéma se situe en dehors de cette plage sont rejetés.
- La taille des échantillons d'histogrammes natifs ne doit pas dépasser 2 048 octets. Les échantillons qui dépassent cette limite sont jetés.

Référence d'API Amazon Managed Service pour Prometheus

Amazon Managed Service for Prometheus propose deux types de : APIs

1. Amazon Managed Service for APIs Prometheus : ils vous APIs permettent de créer et de gérer vos espaces de travail Amazon Managed Service for Prometheus, y compris les opérations relatives aux espaces de travail, aux scrapers, aux définitions du gestionnaire d'alertes, aux espaces de noms des groupes de règles et à la journalisation. Vous utilisez le AWS SDKs, disponible pour différents langages de programmation, pour interagir avec ceux-ci APIs.
2. Compatible avec Prometheus APIs — Amazon Managed Service for Prometheus prend en charge les protocoles HTTP compatibles avec Prometheus. APIs Ils APIs permettent de créer des applications personnalisées, d'automatiser les flux de travail, de s'intégrer à d'autres services ou outils, et d'interroger et d'interagir avec vos données de surveillance à l'aide du langage de requête Prometheus (ProMQL).

Cette section répertorie les opérations et structures de données des API prises en charge par Amazon Managed Service for Prometheus.

Pour plus d'informations sur les quotas pour les séries, les étiquettes et les demandes d'API, consultez les quotas de service Amazon Managed Service for Prometheus dans le guide de l'utilisateur d'Amazon Managed Service for Prometheus.

Rubriques

- Amazon Managed Service pour Prometheus APIs
- Prometheus-compatible APIs

Amazon Managed Service pour Prometheus APIs

Amazon Managed Service for Prometheus fournit des opérations d'API permettant de créer et de gérer vos espaces de travail Amazon Managed Service for Prometheus. Cela inclut APIs les espaces de travail, les scrapers, les définitions du gestionnaire d'alertes, les groupes de règles, les espaces de noms et la journalisation.

Pour obtenir des informations détaillées sur l'Amazon Managed Service for APIs Prometheus, consultez le manuel Amazon Managed Service for Prometheus API Reference.

Utilisation d'Amazon Managed Service pour Prometheus avec un SDK AWS

AWS des kits de développement logiciel (SDKs) sont disponibles pour de nombreux langages de programmation courants. Chaque SDK fournit une API, des exemples de code et une documentation qui permettent aux développeurs de créer plus facilement AWS des applications dans leur langue préférée. Pour une liste des outils par langue SDKs et des outils, consultez la section Outils sur lesquels vous pouvez vous appuyer AWS dans le AWS Developer Center.

> **Versions du SDK**
>
> Nous vous recommandons d'utiliser la version la plus récente du AWS SDK, ainsi que toute autre SDKs version que vous utilisez dans vos projets, et de la maintenir SDKs à jour. Le AWS SDK vous fournit les fonctionnalités les plus récentes, ainsi que des mises à jour de sécurité.

Prometheus-compatible APIs

Amazon Managed Service for Prometheus prend en charge les API suivantes. Prometheus-compatible

Pour plus d'informations sur l'utilisation Prometheus-compatible des API, consultezRequête à l'aide d' Prometheus-compatible API.

Rubriques

- CreateAlertManagerAlerts
- DeleteAlertManagerSilence
- GetAlertManagerStatus
- GetAlertManagerSilence
- GetLabels
- GetMetricMetadata
- GetSeries
- ListAlerts
- ListAlertManagerAlerts

- ListAlertManagerAlertGroups
- ListAlertManagerReceivers
- ListAlertManagerSilences
- ListRules
- PutAlertManagerSilences
- QueryMetrics
- RemoteWrite

CreateAlertManagerAlerts

L'opération `CreateAlertManagerAlerts` crée une alerte dans l'espace de travail.

Verbes HTTP valides :

`POST`

URI valides :

`/workspaces/workspaceId/alertmanager/api/v2/alerts`

Paramètres de requête d'URL :

`alerts` Tableau d'objets, dans lequel chaque objet représente une alerte. Voici un exemple d'objet d'alerte :

```
[
  {
    "startsAt": "2021-09-24T17:14:04.995Z",
    "endsAt": "2021-09-24T17:14:04.995Z",
    "annotations": {
      "additionalProp1": "string",
      "additionalProp2": "string",
      "additionalProp3": "string"
    },
    "labels": {
      "additionalProp1": "string",
      "additionalProp2": "string",
      "additionalProp3": "string"
    },
    "generatorURL": "string"
  }
```

```
]
```

Exemple de demande

```
POST /workspaces/ws-b226cc2a-a446-46a9-933a-ac50479a5568/alertmanager/api/v2/alerts
 HTTP/1.1
Content-Length: 203,
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: Grafana/8.1.0

[
  {
    "labels": {
      "alertname": "test-alert"
    },
    "annotations": {
      "summary": "this is a test alert used for demo purposes"
    },
    "generatorURL": "https://www.amazon.com/"
  }
]
```

Exemple de réponse

```
HTTP/1.1 200 OK
x-amzn-RequestId: 12345678-abcd-4442-b8c5-262b45e9b535
Content-Length: 0
Connection: keep-alive
Date: Tue, 01 Dec 2020 19:37:25 GMT
Content-Type: application/json
Server: amazon
vary: Origin
```

DeleteAlertManagerSilence

`DeleteSilence` supprime un silence d'alerte.

Verbes HTTP valides :

`DELETE`

URI valides :

`/workspaces/workspaceId/alertmanager/api/v2/silence/silenceID`

Paramètres de requête d'URL : aucun

Exemple de demande

```
DELETE /workspaces/ws-b226cc2a-a446-46a9-933a-ac50479a5568/alertmanager/api/v2/silence/
d29d9df3-9125-4441-912c-70b05f86f973 HTTP/1.1
Content-Length: 0,
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: Grafana/8.1.0
```

Exemple de réponse

```
HTTP/1.1 200 OK
x-amzn-RequestId: 12345678-abcd-4442-b8c5-262b45e9b535
Content-Length: 0
Connection: keep-alive
Date: Tue, 01 Dec 2020 19:37:25 GMT
Content-Type: application/json
Server: amazon
vary: Origin
```

GetAlertManagerStatus

`GetAlertManagerStatus` récupère des informations sur le statut du gestionnaire d'alertes.

Verbes HTTP valides :

`GET`

URI valides :

`/workspaces/workspaceId/alertmanager/api/v2/status`

Paramètres de requête d'URL : aucun

Exemple de demande

```
GET /workspaces/ws-b226cc2a-a446-46a9-933a-ac50479a5568/alertmanager/api/v2/status
 HTTP/1.1
Content-Length: 0,
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: Grafana/8.1.0
```

Exemple de réponse

```
HTTP/1.1 200 OK
x-amzn-RequestId: 12345678-abcd-4442-b8c5-262b45e9b535
Content-Length: 941
Connection: keep-alive
Date: Tue, 01 Dec 2020 19:37:25 GMT
Content-Type: application/json
Server: amazon
vary: Origin

{
    "cluster": null,
    "config": {
        "original": "global:\n  resolve_timeout: 5m\n  http_config:\n
 follow_redirects: true\n  smtp_hello: localhost\n  smtp_require_tls: true\nroute:
\n  receiver: sns-0\n  group_by:\n  - label\n  continue: false\nreceivers:\n-
 name: sns-0\n  sns_configs:\n  - send_resolved: false\n    http_config:\n
      follow_redirects: true\n    sigv4: {}\n    topic_arn: arn:aws:sns:us-
west-2:123456789012:test\n    subject: '{{ template \"sns.default.subject\" . }}'\n
    message: '{{ template \"sns.default.message\" . }}'\n    workspace_arn:
 arn:aws:aps:us-west-2:123456789012:workspace/ws-58a6a446-5ec4-415b-9052-a449073bbd0a
\ntemplates: []\n"
    },
    "uptime": null,
    "versionInfo": null
}
```

GetAlertManagerSilence

`GetAlertManagerSilence` récupère des informations sur un silence d'alerte.

Verbes HTTP valides :

- `GET`

URI valides :

```
/workspaces/workspaceId/alertmanager/api/v2/silence/silenceID
```

Paramètres de requête d'URL : aucun

Exemple de demande

```
GET /workspaces/ws-b226cc2a-a446-46a9-933a-ac50479a5568/alertmanager/api/v2/silence/
d29d9df3-9125-4441-912c-70b05f86f973 HTTP/1.1
Content-Length: 0,
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: Grafana/8.1.0
```

Exemple de réponse

```
HTTP/1.1 200 OK
x-amzn-RequestId: 12345678-abcd-4442-b8c5-262b45e9b535
Content-Length: 310
Connection: keep-alive
Date: Tue, 01 Dec 2020 19:37:25 GMT
Content-Type: application/json
Server: amazon
vary: Origin

{
    "id": "d29d9df3-9125-4441-912c-70b05f86f973",
    "status": {
        "state": "active"
    },
    "updatedAt": "2021-10-22T19:32:11.763Z",
    "comment": "hello-world",
    "createdBy": "test-person",
    "endsAt": "2023-07-24T01:05:36.000Z",
    "matchers": [
        {
            "isEqual": true,
            "isRegex": true,
            "name": "job",
            "value": "hello"
        }
```

```
    ],
    "startsAt": "2021-10-22T19:32:11.763Z"
}
```

GetLabels

L'opération `GetLabels` récupère les étiquettes associées à une série temporelle.

Verbes HTTP valides :

`GET`, `POST`

URI valides :

`/workspaces/`*`workspaceId`*`/api/v1/labels`

`/workspaces/`*`workspaceId`*`/api/v1/label/`*`label-name`*`/values` Cette URI ne prend en charge que les requêtes GET.

Paramètres de requête d'URL :

`match[]=<series_selector>` Argument de sélecteur de série répété qui sélectionne la série à partir de laquelle lire les noms d'étiquette. Facultatif.

`start=<rfc3339 | unix_timestamp>` Horodatage de départ. Facultatif.

`end=<rfc3339 | unix_timestamp>` Horodatage de fin. Facultatif.

Exemple de requête pour **`/workspaces/workspaceId/api/v1/labels`**

```
GET /workspaces/ws-b226cc2a-a446-46a9-933a-ac50479a5568/api/v1/labels HTTP/1.1
Content-Length: 0,
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: Grafana/8.1.0
```

Exemple de réponse pour **`/workspaces/workspaceId/api/v1/labels`**

```
HTTP/1.1 200 OK
x-amzn-RequestId: 12345678-abcd-4442-b8c5-262b45e9b535
Content-Length: 1435
```

```
Connection: keep-alive
Date: Tue, 01 Dec 2020 19:37:25 GMT
Content-Type: application/json
Server: amazon
vary: Origin

{
    "status": "success",
    "data": [
        "__name__",
        "access_mode",
        "address",
        "alertname",
        "alertstate",
        "apiservice",
        "app",
        "app_kubernetes_io_instance",
        "app_kubernetes_io_managed_by",
        "app_kubernetes_io_name",
        "area",
        "beta_kubernetes_io_arch",
        "beta_kubernetes_io_instance_type",
        "beta_kubernetes_io_os",
        "boot_id",
        "branch",
        "broadcast",
        "buildDate",
        ...
    ]
}
```

Exemple de requête pour **`/workspaces/workspaceId/api/v1/label/label-name/values`**

```
GET /workspaces/ws-b226cc2a-a446-46a9-933a-ac50479a5568/api/v1/label/access_mode/values
 HTTP/1.1
Content-Length: 0,
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: Grafana/8.1.0
```

Exemple de réponse pour **`/workspaces/workspaceId/api/v1/label/label-name/values`**

```
HTTP/1.1 200 OK
```

```
x-amzn-RequestId: 12345678-abcd-4442-b8c5-262b45e9b535
Content-Length: 74
Connection: keep-alive
Date: Tue, 01 Dec 2020 19:37:25 GMT
Content-Type: application/json
Server: amazon
vary: Origin

{
    "status": "success",
    "data": [
        "ReadWriteOnce"
    ]
}
```

GetMetricMetadata

L'opération `GetMetricMetadata` récupère des métadonnées sur les métriques actuellement extraites des cibles. Elle ne fournit aucune information cible.

La section des données du résultat de la requête se compose d'un objet dans lequel chaque clé est un nom de métrique et chaque valeur est une liste d'objets de métadonnées uniques, telles qu'exposées pour ce nom de métrique sur toutes les cibles.

Verbes HTTP valides :

`GET`

URI valides :

`/workspaces/`*`workspaceId`*`/api/v1/metadata`

Paramètres de requête d'URL :

`limit=<number>` Nombre maximal de métriques à renvoyer.

`metric=<string>` Nom de la métrique pour laquelle filtrer les métadonnées. Si vous laissez ce champ vide, toutes les métadonnées de métriques sont récupérées.

Exemple de demande

```
GET /workspaces/ws-b226cc2a-a446-46a9-933a-ac50479a5568/api/v1/metadata HTTP/1.1
```

```
Content-Length: 0,
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: Grafana/8.1.0
```

Exemple de réponse

```
HTTP/1.1 200 OK
x-amzn-RequestId: 12345678-abcd-4442-b8c5-262b45e9b535
Connection: keep-alive
Date: Tue, 01 Dec 2020 19:37:25 GMT
Content-Type: application/json
Server: amazon
Transfer-Encoding: chunked

{
    "status": "success",
    "data": {
        "aggregator_openapi_v2_regeneration_count": [
            {
                "type": "counter",
                "help": "[ALPHA] Counter of OpenAPI v2 spec regeneration count broken
 down by causing APIService name and reason.",
                "unit": ""
            }
        ],
        ...
    }
}
```

GetSeries

L'opération `GetSeries` récupère la liste des séries temporelles qui correspondent à un ensemble d'étiquettes donné.

Verbes HTTP valides :

- `GET`, `POST`

URI valides :

- `/workspaces/workspaceId/api/v1/series`

Paramètres de requête d'URL :

`match[]=<series_selector>` Argument de sélecteur de série répété qui sélectionne la série à renvoyer. Vous devez fournir au moins un argument `match[]`.

`start=<rfc3339 | unix_timestamp>` Horodatage de départ. Facultatif

`end=<rfc3339 | unix_timestamp>` Horodatage de fin. Facultatif

Exemple de demande

```
POST /workspaces/ws-b226cc2a-a446-46a9-933a-ac50479a5568/api/v1/series --data-urlencode
 'match[]=node_cpu_seconds_total{app="prometheus"}' --data-urlencode 'start=1634936400'
 --data-urlencode 'end=1634939100' HTTP/1.1
Content-Length: 0,
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: Grafana/8.1.0
```

Exemple de réponse

```
HTTP/1.1 200 OK
x-amzn-RequestId: 12345678-abcd-4442-b8c5-262b45e9b535
Connection: keep-alive
Date: Tue, 01 Dec 2020 19:37:25 GMT
Content-Type: application/json
Server: amazon
content-encoding: gzip

{
    "status": "success",
    "data": [
        {
            "__name__": "node_cpu_seconds_total",
            "app": "prometheus",
            "app_kubernetes_io_managed_by": "Helm",
            "chart": "prometheus-11.12.1",
            "cluster": "cluster-1",
            "component": "node-exporter",
            "cpu": "0",
            "heritage": "Helm",
            "instance": "10.0.100.36:9100",
```

```
            "job": "kubernetes-service-endpoints",
            "kubernetes_name": "servicesstackprometheuscf14a6d7-node-exporter",
            "kubernetes_namespace": "default",
            "kubernetes_node": "ip-10-0-100-36.us-west-2.compute.internal",
            "mode": "idle",
            "release": "servicesstackprometheuscf14a6d7"
        },
        {
            "__name__": "node_cpu_seconds_total",
            "app": "prometheus",
            "app_kubernetes_io_managed_by": "Helm",
            "chart": "prometheus-11.12.1",
            "cluster": "cluster-1",
            "component": "node-exporter",
            "cpu": "0",
            "heritage": "Helm",
            "instance": "10.0.100.36:9100",
            "job": "kubernetes-service-endpoints",
            "kubernetes_name": "servicesstackprometheuscf14a6d7-node-exporter",
            "kubernetes_namespace": "default",
            "kubernetes_node": "ip-10-0-100-36.us-west-2.compute.internal",
            "mode": "iowait",
            "release": "servicesstackprometheuscf14a6d7"
        },
        ...
    ]
}
```

ListAlerts

L'opération `ListAlerts` récupère les alertes actuellement actives dans l'espace de travail.

Verbes HTTP valides :

`GET`

URI valides :

`/workspaces/workspaceId/api/v1/alerts`

Exemple de demande

```
GET /workspaces/ws-b226cc2a-a446-46a9-933a-ac50479a5568/api/v1/alerts HTTP/1.1
```

```
Content-Length: 0,
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: Grafana/8.1.0
```

Exemple de réponse

```
HTTP/1.1 200 OK
x-amzn-RequestId: 12345678-abcd-4442-b8c5-262b45e9b535
Content-Length: 386
Connection: keep-alive
Date: Tue, 01 Dec 2020 19:37:25 GMT
Content-Type: application/json
Server: amazon
vary: Origin

{
  "status": "success",
  "data": {
    "alerts": [
      {
        "labels": {
          "alertname": "test-1.alert",
          "severity": "none"
        },
        "annotations": {
          "message": "message"
        },
        "state": "firing",
        "activeAt": "2020-12-01T19:37:25.429565909Z",
        "value": "1e+00"
      }
    ]
  },
  "errorType": "",
  "error": ""
}
```

ListAlertManagerAlerts

`ListAlertManagerAlerts` récupère des informations sur les alertes en cours dans le gestionnaire d'alertes de l'espace de travail.

Verbes HTTP valides :

`GET`

URI valides :

`/workspaces/workspaceId/alertmanager/api/v2/alerts`

Paramètres de requête d'URL :

Booléen `active`. Si la valeur est true, la liste renvoyée inclut les alertes actives. Par défaut, la valeur est true. Facultatif

Booléen `silenced`. Si la valeur est true, la liste renvoyée inclut les alertes silencieuses. Par défaut, la valeur est true. Facultatif

Booléen `inhibited`. Si la valeur est true, la liste renvoyée inclut les alertes bloquées. Par défaut, la valeur est true. Facultatif

`filter` Tableau de chaînes. Liste d'analyseurs permettant de filtrer les alertes. Facultatif

`receiver` Chaîne. Expression régulière correspondant aux récepteurs pour filtrer les alertes. Facultatif

Exemple de demande

```
GET /workspaces/ws-b226cc2a-a446-46a9-933a-ac50479a5568/alertmanager/api/v2/alerts
 HTTP/1.1
Content-Length: 0,
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: Grafana/8.1.0
```

Exemple de réponse

```
HTTP/1.1 200 OK
x-amzn-RequestId: 12345678-abcd-4442-b8c5-262b45e9b535
Content-Length: 354
Connection: keep-alive
Date: Tue, 01 Dec 2020 19:37:25 GMT
Content-Type: application/json
Server: amazon
```

```
vary: Origin

[
    {
        "annotations": {
            "summary": "this is a test alert used for demo purposes"
        },
        "endsAt": "2021-10-21T22:07:31.501Z",
        "fingerprint": "375eab7b59892505",
        "receivers": [
            {
                "name": "sns-0"
            }
        ],
        "startsAt": "2021-10-21T22:02:31.501Z",
        "status": {
            "inhibitedBy": [],
            "silencedBy": [],
            "state": "active"
        },
        "updatedAt": "2021-10-21T22:02:31.501Z",
        "labels": {
            "alertname": "test-alert"
        }
    }
]
```

ListAlertManagerAlertGroups

L'opération `ListAlertManagerAlertGroups` récupère la liste des groupes d'alertes configurés dans le gestionnaire d'alertes de l'espace de travail.

Verbes HTTP valides :

`GET`

URI valides :

`/workspaces/workspaceId/alertmanager/api/v2/alerts/groups`

Paramètres de requête d'URL :

Booléen `active`. Si la valeur est true, la liste renvoyée inclut les alertes actives. Par défaut, la valeur est true. Facultatif

Booléen `silenced`. Si la valeur est true, la liste renvoyée inclut les alertes silencieuses. Par défaut, la valeur est true. Facultatif

Booléen `inhibited`. Si la valeur est true, la liste renvoyée inclut les alertes bloquées. Par défaut, la valeur est true. Facultatif

Booléen `muted`. Si c'est vrai, la liste renvoyée inclut les alertes désactivées. Par défaut, la valeur est true. Facultatif

`filter` Tableau de chaînes. Liste d'analyseurs permettant de filtrer les alertes. Facultatif

`receiver` Chaîne. Expression régulière correspondant aux récepteurs pour filtrer les alertes. Facultatif

Exemple de demande

```
GET /workspaces/ws-b226cc2a-a446-46a9-933a-ac50479a5568/alertmanager/api/v2/alerts/
groups HTTP/1.1
Content-Length: 0,
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: Grafana/8.1.0
```

Exemple de réponse

```
HTTP/1.1 200 OK
x-amzn-RequestId: 12345678-abcd-4442-b8c5-262b45e9b535
Content-Length: 443
Connection: keep-alive
Date: Tue, 01 Dec 2020 19:37:25 GMT
Content-Type: application/json
Server: amazon
vary: Origin

[
    {
        "alerts": [
            {
                "annotations": {
                    "summary": "this is a test alert used for demo purposes"
                },
```

```
                "endsAt": "2021-10-21T22:07:31.501Z",
                "fingerprint": "375eab7b59892505",
                "receivers": [
                    {
                        "name": "sns-0"
                    }
                ],
                "startsAt": "2021-10-21T22:02:31.501Z",
                "status": {
                    "inhibitedBy": [],
                    "silencedBy": [],
                    "state": "unprocessed"
                },
                "updatedAt": "2021-10-21T22:02:31.501Z",
                "generatorURL": "https://www.amazon.com/",
                "labels": {
                    "alertname": "test-alert"
                }
            }
        ],
        "labels": {},
        "receiver": {
            "name": "sns-0"
        }
    }
]
```

ListAlertManagerReceivers

L'opération `ListAlertManagerReceivers` récupère des informations sur les récepteurs configurés dans le gestionnaire d'alertes.

Verbes HTTP valides :

`GET`

URI valides :

`/workspaces/workspaceId/alertmanager/api/v2/receivers`

Paramètres de requête d'URL : aucun

Exemple de demande

```
GET /workspaces/ws-b226cc2a-a446-46a9-933a-ac50479a5568/alertmanager/api/v2/receivers
 HTTP/1.1
Content-Length: 0,
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: Grafana/8.1.0
```

Exemple de réponse

```
HTTP/1.1 200 OK
x-amzn-RequestId: 12345678-abcd-4442-b8c5-262b45e9b535
Content-Length: 19
Connection: keep-alive
Date: Tue, 01 Dec 2020 19:37:25 GMT
Content-Type: application/json
Server: amazon
vary: Origin

[
    {
        "name": "sns-0"
    }
]
```

ListAlertManagerSilences

L'opération `ListAlertManagerSilences` récupère des informations sur les silences d'alerte configurés dans l'espace de travail.

Verbes HTTP valides :

`GET`

URI valides :

`/workspaces/`*`workspaceId`*`/alertmanager/api/v2/silences`

Paramètres de requête d'URL :

`filter` Tableau de chaînes. Une liste de facteurs permettant de filtrer les silences. Facultatif

Exemple de demande

```
GET /workspaces/ws-58a6a446-5ec4-415b-9052-a449073bbd0a/alertmanager/api/v2/silences
 HTTP/1.1
Content-Length: 0,
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: Grafana/8.1.0
```

Exemple de réponse

```
HTTP/1.1 200 OK
x-amzn-RequestId: 12345678-abcd-4442-b8c5-262b45e9b535
Content-Length: 312
Connection: keep-alive
Date: Tue, 01 Dec 2020 19:37:25 GMT
Content-Type: application/json
Server: amazon
vary: Origin

[
    {
        "id": "d29d9df3-9125-4441-912c-70b05f86f973",
        "status": {
            "state": "active"
        },
        "updatedAt": "2021-10-22T19:32:11.763Z",
        "comment": "hello-world",
        "createdBy": "test-person",
        "endsAt": "2023-07-24T01:05:36.000Z",
        "matchers": [
            {
                "isEqual": true,
                "isRegex": true,
                "name": "job",
                "value": "hello"
            }
        ],
        "startsAt": "2021-10-22T19:32:11.763Z"
    }
]
```

ListRules

L'`ListRules`opération récupère des informations sur les règles configurées dans l'espace de travail.

Une seule réponse contient au maximum 100 000 règles. Comme les groupes de règles ne sont jamais répartis sur plusieurs pages, la troncature atteint toujours une limite de groupe de règles. Une réponse peut donc contenir moins de 100 000 règles si l'inclusion du groupe de règles suivant dépasse le plafond. En cas de troncature, la réponse inclut la `groupNextToken` propriété. Pour récupérer la page suivante, transmettez cette valeur en tant que paramètre de `group_next_token` requête, que vous devez envoyer avec`group_limit`.

Verbes HTTP valides :

- `GET`

URI valides :

- `/workspaces/`*`workspaceId`*`/api/v1/rules`

Paramètres de requête d'URL :

- `type=<alert|record>`Renvoie uniquement les règles d'alerte (par exemple,`type=alert`) ou les règles d'enregistrement (par exemple,`type=record`). Lorsque le paramètre est absent ou vide, aucun filtrage n'est effectué. Facultatif

- `rule_name[]=<string>`Filtre les règles en utilisant les noms fournis. Lorsque plusieurs filtres sont fournis, le nom de la règle doit satisfaire au moins l'un des filtres indiqués. Facultatif

- `rule_group[]=<string>`Filtre les groupes de règles en utilisant les noms fournis. Lorsque plusieurs filtres sont fournis, le nom du groupe de règles doit satisfaire au moins l'un des filtres indiqués. Facultatif

- `file[]=<string>`Filtre les espaces de noms des groupes de règles en utilisant les noms fournis. Lorsque plusieurs filtres sont fournis, le nom de l'espace de noms du groupe de règles doit satisfaire au moins l'un des filtres indiqués. Facultatif

- `exclude_alerts=<true|false>`Filtre les règles sans aucune alerte active lorsqu'elle est définie sur`true`. Facultatif

- `match[]=<series_selector>`Renvoie uniquement les règles qui ont configuré des étiquettes conformes aux sélecteurs d'étiquettes. Si le paramètre est répété, les règles correspondant à l'un des ensembles de sélecteurs d'étiquettes sont renvoyées. Notez que la correspondance se

fait sur les étiquettes de la définition de chaque règle, et non sur les valeurs après l'extension du modèle (pour les règles d'alerte). Facultatif

`group_limit=<number>`Nombre maximal de groupes de règles renvoyés dans une seule réponse. Dans les demandes de pagination suivantes, utilisez la `groupNextToken` propriété de cette réponse pour définir le `group_next_token` paramètre. Facultatif

`group_next_token=<string>`Le jeton de pagination renvoyé par une demande précédente lorsque vous définissez la `group_limit` propriété ou lorsque la réponse atteint le plafond de 100 000 règles. Utilisez ce jeton pour paginer de manière itérative sur les groupes de règles restants. Lorsque vous utilisez ce paramètre, vous devez également envoyer le `group_limit` paramètre. Facultatif

Exemple de demande

```
GET /workspaces/ws-b226cc2a-a446-46a9-933a-ac50479a5568/api/v1/rules HTTP/1.1
Content-Length: 0,
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: Grafana/8.1.0
```

Exemple de réponse

```
HTTP/1.1 200 OK
x-amzn-RequestId: 12345678-abcd-4442-b8c5-262b45e9b535
Content-Length: 423
Connection: keep-alive
Date: Tue, 01 Dec 2020 19:37:25 GMT
Content-Type: application/json
Server: amazon
vary: Origin

{
    "status": "success",
    "data": {
        "groups": [
            {
                "name": "test-1.rules",
                "file": "test-rules",
                "rules": [
                    {
```

```
                        "name": "record:1",
                        "query": "sum(rate(node_cpu_seconds_total[10m:1m]))",
                        "labels": {},
                        "health": "ok",
                        "lastError": "",
                        "type": "recording",
                        "lastEvaluation": "2021-10-21T21:22:34.429565909Z",
                        "evaluationTime": 0.001005399
                    }
                ],
                "interval": 60,
                "lastEvaluation": "2021-10-21T21:22:34.429563992Z",
                "evaluationTime": 0.001010504
            }
        ]
    },
    "errorType": "",
    "error": ""
}
```

PutAlertManagerSilences

L'opération `PutAlertManagerSilences` crée un nouveau silence d'alerte ou met à jour un silence existant.

Verbes HTTP valides :

`POST`

URI valides :

`/workspaces/workspaceId/alertmanager/api/v2/silences`

Paramètres de requête d'URL :

`silence` Objet qui représente le silence. En voici le format :

```
{
  "id": "string",
  "matchers": [
    {
      "name": "string",
      "value": "string",
      "isRegex": Boolean,
```

```
      "isEqual": Boolean
    }
  ],
  "startsAt": "timestamp",
  "endsAt": "timestamp",
  "createdBy": "string",
  "comment": "string"
}
```

Exemple de demande

```
POST /workspaces/ws-b226cc2a-a446-46a9-933a-ac50479a5568/alertmanager/api/v2/silences
 HTTP/1.1
Content-Length: 281,
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: Grafana/8.1.0

{
   "matchers":[
      {
         "name":"job",
         "value":"up",
         "isRegex":false,
         "isEqual":true
      }
   ],
   "startsAt":"2020-07-23T01:05:36+00:00",
   "endsAt":"2023-07-24T01:05:36+00:00",
   "createdBy":"test-person",
   "comment":"test silence"
}
```

Exemple de réponse

```
HTTP/1.1 200 OK
x-amzn-RequestId: 12345678-abcd-4442-b8c5-262b45e9b535
Content-Length: 53
Connection: keep-alive
Date: Tue, 01 Dec 2020 19:37:25 GMT
Content-Type: application/json
Server: amazon
```

```
vary: Origin

{
    "silenceID": "512860da-74f3-43c9-8833-cec026542b32"
}
```

QueryMetrics

L'opération `QueryMetrics` évalue une requête instantanée à un moment donné ou sur une période donnée.

Verbes HTTP valides :

- `GET`, `POST`

URI valides :

- `/workspaces/`*`workspaceId`*`/api/v1/query` Cet URI évalue une requête instantanée à un moment donné.
- `/workspaces/`*`workspaceId`*`/api/v1/query_range` Cet URI évalue une requête instantanée sur une période donnée.

Paramètres de requête d'URL :

- `query=<string>` Chaîne de requête d'expression Prometheus. Utilisée à la fois dans `query` et `query_range`.
- `time=<rfc3339 | unix_timestamp>` (Facultatif) Horodatage d'évaluation si vous utilisez `query` pour une requête instantanée à un moment donné.
- `timeout=<duration>` (Facultatif) Délai d'évaluation. La valeur par défaut est définie et plafonnée par la valeur de l'indicateur `-query.timeout`. Utilisée à la fois dans `query` et `query_range`.
- `start=<rfc3339 | unix_timestamp>` Démarrez l'horodatage si vous utilisez `query_range` pour effectuer des requêtes sur un intervalle de temps.
- `end=<rfc3339 | unix_timestamp>` Arrêtez l'horodatage si vous utilisez `query_range` pour effectuer des requêtes sur un intervalle de temps.
- `step=<duration | float>` Interrogez la durée de l'étape de résolution sous forme de `duration` ou sous forme d'un nombre `float` de secondes. À utiliser uniquement si vous utilisez

`query_range` pour effectuer des requêtes sur un intervalle de temps, et nécessaire pour ce type de requêtes.

`max_samples_processed_warning_threshold=<integer>`(Facultatif) Définit le seuil d'avertissement pour les échantillons de requêtes traités (QSP). Lorsque les requêtes atteignent ce seuil, un message d'avertissement est renvoyé dans la réponse de l'API.

`max_samples_processed_error_threshold=<integer>`>(Facultatif) Définit le seuil d'erreur pour les échantillons de requêtes traités (QSP). Les requêtes dépassant ce seuil seront rejetées avec une erreur et ne seront pas facturées. Utilisé pour éviter des coûts de requête excessifs.

Durée

`duration`Dans une Prometheus-compatible API, un nombre est immédiatement suivi de l'une des unités suivantes :

- `ms` millisecondes
- `s` secondes
- `m` minutes
- `h` heures
- `d` jours, en supposant qu'un jour compte toujours 24 heures
- `w` semaines, en supposant qu'une semaine compte toujours 7 jours
- `y` années, en supposant qu'une année compte toujours 365 jours

Exemple de demande

```
POST /workspaces/ws-b226cc2a-a446-46a9-933a-ac50479a5568/api/v1/query?
query=sum(node_cpu_seconds_total) HTTP/1.1
Content-Length: 0,
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: Grafana/8.1.0
```

Exemple de réponse

```
HTTP/1.1 200 OK
x-amzn-RequestId: 12345678-abcd-4442-b8c5-262b45e9b535
```

```
Content-Length: 132
Connection: keep-alive
Date: Tue, 01 Dec 2020 19:37:25 GMT
Content-Type: application/json
Server: amazon
content-encoding: gzip

{
    "status": "success",
    "data": {
        "resultType": "vector",
        "result": [
            {
                "metric": {},
                "value": [
                    1634937046.322,
                    "252590622.81000024"
                ]
            }
        ]
    }
}
```

RemoteWrite

L'opération `RemoteWrite` écrit les métriques d'un serveur Prometheus sur une URL distante dans un format normalisé. Généralement, vous utilisez un client existant tel qu'un serveur Prometheus pour appeler cette opération.

Verbes HTTP valides :

`POST`

URI valides :

`/workspaces/workspaceId/api/v1/remote_write`

Paramètres de requête d'URL :

Aucune

`RemoteWrite` a un taux d'ingestion de 70 000 échantillons par seconde et une taille de rafale d'ingestion de 1 000 000 échantillons.

Exemple de demande

```
POST /workspaces/ws-b226cc2a-a446-46a9-933a-ac50479a5568/api/v1/remote_write --data-
binary "@real-dataset.sz" HTTP/1.1
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: Prometheus/2.20.1
Content-Type: application/x-protobuf
Content-Encoding: snappy
X-Prometheus-Remote-Write-Version: 0.1.0

body
```

> **Note**
>
> Pour la syntaxe du corps de la requête, reportez-vous à la définition de la mémoire tampon du protocole à l'https://github.com/prometheus/prometheus/blob/1c624c58ca934f618be737b4995e22051f5724c1/prompb/remote.pb.goadresse #L64.

Exemple de réponse

```
HTTP/1.1 200 OK
x-amzn-RequestId: 12345678-abcd-4442-b8c5-262b45e9b535
Content-Length:0
Connection: keep-alive
Date: Tue, 01 Dec 2020 19:37:25 GMT
Content-Type: application/json
Server: amazon
vary: Origin
```

Historique du Guide de l’utilisateur Amazon Managed Service for Prometheus

Le tableau suivant décrit les mises à jour importantes de documentation dans le Guide de l’utilisateur Amazon Managed Service for Prometheus. Pour recevoir les notifications de mise à jour de cette documentation, abonnez-vous à un flux RSS.

Modification	Description	Date
Mise à jour de la AmazonPrometheusScraperServiceRolePolicy politique IAM gérée	AWS les collecteurs gérés peuvent désormais écrire les métriques collectées dans les ensembles de données Amazon CloudWatch Metrics en tant que nouveau type de destination.	14 juillet 2026
Ajout du support natif des histogrammes	Amazon Managed Service for Prometheus prend désormais en charge l'ingestion, le stockage et l'interrogation des histogrammes natifs de Prometheus. De nouvelles mesures CloudWatch `NativeHistogramIngestionRate` d'utilisation `NativeHistogramActiveSeries` sont disponibles pour surveiller l'utilisation de l'histogramme natif par rapport aux limites de quotas. Une nouvelle métrique de service fournit une visibilité sur la consommation des compartiments. `NativeHistogramIngestedBuck`	10 juin 2026

	`etsRate` Pour plus d'informations, consultez la section Surveillance à l'aide de CloudWatch métriques.	
Ajout de l'ingestion hors commande et de la prise en charge du décalage des requêtes de règles	Amazon Managed Service for Prometheus prend désormais en charge la configuration d'une fenêtre temporelle en cas de rupture de commande pour accepter les échantillons qui arrivent hors ordre chronologique, et d'un décalage des requêtes de règles pour introduire un délai global avant l'évaluation des règles. Deux nouvelles CloudWatch mesures, `OutOfOrderIngestionRate` et`OutOfOrderSampleAge` , sont également disponibles pour surveiller l'ingestion de produits hors commande. Pour plus d'informations, consultez Configurer votre espace de travail.	14 mai 2026

Support lancé pour PagerDuty	Amazon Managed Service for Prometheus ajoute la prise en charge de l'intégration qui permet d'automatiser PagerDuty les flux de travail de réponse aux incidents et de garantir que les alertes critiques parviennent aux bons membres de l'équipe au bon moment. Pour plus d'informations, voir Utiliser PagerDuty comme récepteur d'alertes.	29 août 2025
Ajout d'un support politique basé sur les ressources	Les actions d'API suivantes sont désormais disponibles : • DeleteResourcePolicy • DescribeResourcePolicy • PutResourcePolicy	15 août 2025
Mise à jour de la politique IAM AmazonPrometheusConsoleFullAccess gérée.	La AmazonPrometheusConsoleFullAccesspolitique a été mise à jour. Les `aps:DescribeQueryLoggingConfiguration` autorisations `aps:CreateQueryLoggingConfiguration` `aps:UpdateQueryLoggingConfiguration` ,`aps:DeleteQueryLoggingConfiguration` , ont été ajoutées à la politique.	5 mai 2025

Ajout de la modification des fichiers de définition des règles et des fichiers de configuration du gestionnaire d'alertes dans la console	Amazon Managed Service for Prometheus permet désormais de modifier les fichiers de configuration du gestionnaire d'alertes et les fichiers de définition des règles depuis la console Amazon Managed Service for Prometheus.	16 mai 2024
Configuration simplifiée du collecteur AWS géré avec entrées d'accès pour Amazon EKS	Amazon Managed Service for Prometheus prend désormais en charge les entrées d'accès Amazon EKS afin de simplifier la configuration AWS des collecteurs gérés. La politique de AmazonPrometheusScraperServiceRolePolicygestion pour les collecteurs AWS gérés est mise à jour pour permettre la suppression des entrées d'accès qui ne sont plus utilisées.	2 mai 2024
Déplacer AWS l'API vers un guide de référence d'API distinct	Les API Amazon Managed Service for AWS Prometheus sont désormais disponibles dans leur propre référence, l'Amazon Managed Service for Prometheus API Reference. Prometheus-compatible Les API continuent d'être documentées dans le guide de l'utilisateur d'Amazon Managed Service for Prometheus.	7 février 2024

Ajout de clés gérées par le client pour le chiffrement de l'espace de travail	Amazon Managed Service for Prometheus ajoute la prise en charge des clés gérées par le client pour le chiffrement de l'espace de travail. Pour plus d'informations, consultez Chiffrement au repos.	21 décembre 2023
Ajout de nouvelles autorisations à AmazonPrometheusFullAccess	De nouvelles autorisations ont été ajoutées à la politique AmazonPrometheusFullAccessgérée afin de permettre la création de collecteurs AWS gérés pour les clusters Amazon EKS.	26 novembre 2023
Ajout d'une nouvelle politique gérée, AmazonPrometheusScraperServiceLinkedRolePolicy	Ajout d'une nouvelle politique gérée, AmazonPrometheusScraperServiceLinkedRolePolicypermettant aux collecteurs AWS gérés de collecter des métriques à partir de clusters Amazon EKS.	26 novembre 2023
Ajout de collecteurs AWS gérés comme méthode d'ingestion	Amazon Managed Service for Prometheus ajoute la prise en charge des collecteurs gérés AWS.	26 novembre 2023
Ajout de la prise en charge de l'intégration à Amazon Managed Grafana	Amazon Managed Service for Prometheus ajoute la prise en charge de l'intégration aux alertes Amazon Managed Grafana.	23 novembre 2022

Ajout de nouvelles autorisations à AmazonPrometheusConsoleFullAccess	De nouvelles autorisations ont été ajoutées à la politique AmazonPrometheusConsoleFullAccessgérée pour prendre en charge la journalisation des événements liés au gestionnaire d'alertes et aux règles dans CloudWatch les journaux.	24 octobre 2022
Ajout de la solution d'observabilité Amazon EKS.	Amazon Managed Service for Prometheus ajoute une nouvelle solution AWS utilisant Observability Accelerator. Pour plus d'informations, consultez la section Using AWS Observability Accelerator.	14 octobre 2022
Ajout de la prise en charge de l'intégration dans le suivi des coûts d'Amazon EKS.	Amazon Managed Service for Prometheus ajoute la prise en charge de l'intégration dans le suivi des coûts d'Amazon EKS. Pour de plus amples informations, consultez la section Integrating with Amazon EKS cost monitoring.	22 septembre 2022
Lancement de la prise en charge des journaux Alert Manager et Ruler dans Amazon CloudWatch Logs.	Amazon Managed Service for Prometheus prend désormais en charge les journaux d'erreurs Alert Manager et Ruler dans Amazon Logs. CloudWatch Pour plus d'informations, consultez Amazon CloudWatch Logs.	1er septembre 2022

Ajout de la prise en charge de la conservation du stockage personnalisée.	Amazon Managed Service for Prometheus ajoute la prise en charge de la conservation du stockage personnalisée par espace de travail, en modifiant le quota de cet espace de travail. Pour plus d'informations sur les quotas dans Amazon Managed Service for Prometheus, consultez la section Service quotas.	12 août 2022
Des statistiques d'utilisation ont été ajoutées à Amazon CloudWatch.	Amazon Managed Service for Prometheus prend désormais en charge l'envoi de statistiques d'utilisation à Amazon. CloudWatch Pour plus d'informations, consultez les CloudWatchmétriques Amazon.	6 mai 2022
Ajout de la prise en charge de la région Europe (Londres).	Amazon Managed Service for Prometheus prend désormais en charge la région Europe (Londres).	4 mai 2022
Amazon Managed Service for Prometheus est généralement disponible et prend désormais en charge les règles et le gestionnaire d'alertes.	Amazon Managed Service for Prometheus est généralement disponible. Les règles et le gestionnaire d'alertes sont également pris en charge. Pour plus d'informations, consultez les sections Règles d'enregistrement et règles d'alerte et Gestionnaire d'alertes et modélisation.	29 septembre 2021

Ajout de la prise en charge du balisage.	Amazon Managed Service for Prometheus prend en charge le balisage des espaces de travail Amazon Managed Service for Prometheus.	7 septembre 2021
Les séries actives et les quotas de taux d'ingestion ont été augmentés.	Le quota de séries actives est passé à 1 000 000 et le quota de taux d'ingestion est passé à 70 000 échantillons par seconde.	22 février 2021
Version préliminaire d'Amazon Managed Service for Prometheus.	La version préliminaire d'Amazon Managed Service for Prometheus est disponible.	15 décembre 2020

Les traductions sont fournies par des outils de traduction automatique. En cas de conflit entre le contenu d'une traduction et celui de la version originale en anglais, la version anglaise prévaudra.