aws

Benutzer-Leitfaden

Amazon Managed Service für Prometheus



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

Amazon Managed Service für Prometheus: Benutzer-Leitfaden

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

Die Handelsmarken und die Handelsaufmachung von Amazon dürfen nicht in einer Weise in Verbindung mit nicht von Amazon stammenden Produkten oder Services verwendet werden, durch die Kunden irregeführt werden könnten oder Amazon in schlechtem Licht dargestellt oder diskreditiert werden könnte. Alle anderen Handelsmarken, die nicht Eigentum von Amazon sind, gehören den jeweiligen Besitzern, die möglicherweise zu Amazon gehören oder nicht, mit Amazon verbunden sind oder von Amazon gesponsert werden.

Table of Contents

Was Amazon Managed Service für Prometheus? 1
Unterstützte Regionen 1
Preisgestaltung 13
Premium-Support 13
Erste Schritte 14
Einrichten AWS 14
Melden Sie sich für eine an AWS-Konto 14
Einen Workspace erstellen 15
Erfassungsmetriken 16
Schritt 1: Hinzufügen neuer Helm-Chart-Repositorys 17
Schritt 2: Erstellen eines Prometheus-Namespace 17
Schritt 3: Einrichten von IAM-Rollen für Servicekonten 17
Schritt 4: Einrichten des neuen Servers und Beginn der Erfassung von Metriken 18
Metriken abfragen 19
Arbeitsbereiche verwalten 21
Einen Workspace erstellen 21
Konfiguriere deinen Workspace 25
Bearbeiten Sie einen Workspace-Alias 27
Finden Sie Ihre Workspace-Details 27
Einen Workspace löschen 30
Erfassungsmetriken 31
AWS verwaltete Sammler 32
Integrieren Sie Amazon EKS 33
Integrieren Sie Amazon MSK 53
Integrieren Sie Amazon OpenSearch Service 84
Prometheus-compatible Metriken 97
Sammler überwachen 98
Kundenverwaltete Kollektoren 104
Sichern Sie die Erfassung Ihrer Metriken 105
ADOT-Kollektoren 105
Prometheus Kollektoren 123
High-availability Daten 133
Fragen Sie Ihre Metriken ab 142
PromQL-Spickzettel 143

Grundlegende Selektoren ... 143
Selektoren für Bereichsvektoren ... 143
Aggregationsoperatoren ... 144
Allgemeine Funktionen ... 144
Binäre Operatoren ... 145
Praktische Beispiele für Abfragen ... 145
Schützen Sie Ihre metrischen Abfragen ... 146
Verwenden AWS PrivateLink mit Amazon Managed Service für Prometheus ... 105
Authentifizierung und Autorisierung ... 105
Verwenden Sie Amazon Managed Grafana ... 147
Verbindung zu Amazon Managed Grafana in einer privaten VPC herstellen ... 147
Verwenden Sie Grafana Open Source ... 148
Voraussetzungen ... 148
Schritt 1: Einrichten von AWS SigV4 ... 149
Schritt 2: Fügen Sie die Prometheus-Datenquelle in Grafana hinzu ... 150
Schritt 3: (optional) Fehlerbehebung, wenn Save Test nicht funktioniert & ... 152
Verwenden Sie Grafana in Amazon EKS ... 153
Einrichten AWS SigV4 ... 154
IAM-Rollen für Servicekonten einstellen ... 154
Aktualisieren Sie den Grafana-Server mit Helm ... 156
Fügen Sie die Prometheus-Datenquelle in Grafana hinzu ... 156
Verwenden Sie direkte Abfragen ... 157
Mit awscurl abfragen ... 158
Statistiken abfragen ... 160
Anomalie-Erkennung ... 164
Funktionsweise der Anomalieerkennung ... 164
Erste Schritte mit der Anomalieerkennung ... 165
PreviewAnomalyDetector ... 165
Formatierung von Abfrageparametern ... 166
API-Anforderung und -Antwort ... 166
Regeln für die Aufzeichnung und Benachrichtigung ... 170
Erforderliche IAM-Berechtigungen ... 171
Erstellen Sie eine Regeldatei ... 172
Laden Sie eine Regeldatei hoch ... 174
Bearbeiten Sie eine Regeldatei ... 176
Problembehandlung bei Regelauswertungen ... 177

Überprüfen Sie den Status der Alarmauslösung ... 178
Beheben Sie fehlende Warnmeldungen ... 178
Überprüfen Sie den Integritätsstatus der Regel ... 179
Verwenden Sie Offset in Abfragen, um Verzögerungen bei der Datenerfassung zu beheben ... 181
Häufige Probleme und Lösungen ... 182
Bewährte Verfahren für die Bewertung von Regeln ... 183
Fehlersuche-Ruler ... 183
Alert Manager ... 185
Erforderliche IAM-Berechtigungen ... 186
Erstellen Sie eine Konfigurationsdatei ... 187
Richten Sie einen Alarmempfänger ein ... 189
Amazon SNS ... 190
PagerDuty ... 201
Laden Sie eine Konfigurationsdatei hoch ... 207
Integrieren Sie Benachrichtigungen mit Grafana ... 210
Voraussetzungen ... 210
Einrichten von Amazon Managed Grafana ... 211
Beheben Sie Fehler im Alert Manager ... 212
Warnung vor aktiven Alerts ... 213
Warnung zur Größe der Alert-Aggregationsgruppe ... 213
Die Größe der Warnmeldungen ist zu groß, Warnung ... 214
Warnung vor leerem Inhalt ... 215
Ungültige `key/value` Warnung ... 215
Warnung vor Nachrichtenlimit ... 216
Kein ressourcenbasierter Richtlinienfehler ... 216
Nicht-ASCII-Warnung ... 217
Nicht berechtigt, KMS anzurufen ... 217
Vorlagenfehler ... 218
Überwachung von Arbeitsbereichen ... 219
CloudWatch Metriken ... 219
Einen Alarm einrichten CloudWatch ... 237
CloudWatch Protokolle ... 238
CloudWatch Protokolle konfigurieren ... 238
Einblicke und Kontrolle über Abfragen ... 241
Konfiguration der Abfrageprotokollierung ... 241

Schwellenwerte für die Abfragedrosselung konfigurieren ... 243
Inhalt protokollieren ... 244
Einschränkungen ... 245
Kosten verstehen und optimieren ... 246
Was trägt zu meinen Kosten bei? ... 246
Wie kann ich meine Kosten am besten senken? Wie senke ich die Kosten für die Erfassung? ... 246
Wie kann ich meine Abfragekosten am besten senken? ... 247
Wenn ich die Aufbewahrungsdauer meiner Metriken verkürze, trägt das dazu bei, meine Gesamtrechnung zu reduzieren? ... 247
Wie kann ich die Kosten für meine Warnungsabfragen niedrig halten? ... 247
Kann ich meine Rechnung jederzeit überprüfen? ... 248
Welche Metriken kann ich verwenden, um meine Kosten zu überwachen? ... 249
Wie sehe ich meine Kosten in AWS Cost Explorer? ... 250
Wie berechne ich die Anzahl der in einem Monat aufgenommenen Proben? ... 251
Welche Datengranularität ist für die Analyse historischer Kosten verfügbar? ... 253
Was sind bewährte Methoden für die Überwachung der Kosten von Amazon Managed Service for Prometheus? ... 254
Warum ist meine Rechnung zu Beginn des Monats höher als am Monatsende? ... 254
Ich habe alle meine Amazon Managed Service for Prometheus-Arbeitsbereiche gelöscht, aber es scheint, dass mir immer noch Gebühren berechnet werden. Was könnte passieren? ... 255
Integrationen ... 256
Amazon-EKS-Kostenüberwachung ... 256
AWS Observability-Beschleuniger ... 257
Voraussetzungen ... 258
Verwenden Sie das Beispiel für verwaltete Metriken (ohne Agenten) ... 258
Alternative: Selbstverwalteter Collector OpenTelemetry ... 260
Dashboards anzeigen ... 261
AWS Controller für Kubernetes ... 261
Voraussetzungen ... 262
Einen Workspace bereitstellen ... 263
Den Cluster für Remote-Write konfigurieren ... 267
CloudWatch Amazon-Metriken mit Firehose ... 269
Infrastruktur ... 270
Einen CloudWatch Amazon-Stream erstellen ... 272
Bereinigen ... 273
Sicherheit ... 275

Datenschutz 276
Von Amazon Managed Service für Prometheus gesammelte Daten 277
Verschlüsselung im Ruhezustand 278
Identitäts- und Zugriffsverwaltung 291
Zielgruppe 292
Authentifizierung mit Identitäten 292
Verwalten des Zugriffs mit Richtlinien 294
So funktioniert Amazon Managed Service für Prometheus mit IAM 296
Beispiele für identitätsbasierte Richtlinien 302
Fehlerbehebung 305
IAM-Berechtigungen und Richtlinien 307
Berechtigungen für Amazon Managed Service für Prometheus 307
Beispiel für IAM-Richtlinien 307
Compliance-Validierung 308
Ausfallsicherheit 309
Infrastruktursicherheit 309
Verwenden von servicegebundenen Rollen 310
Metrik-Scraping-Rolle 310
CloudTrail protokolliert 312
Verwaltungsveranstaltungen von Amazon Managed Service für Prometheus in CloudTrail .. 314
Beispiele für Veranstaltungen mit Amazon Managed Service für Prometheus 315
IAM-Rollen für Servicekonten einrichten 319
Einrichten von Servicerollen für die Erfassung von Metriken aus Amazon-EKS-Clustern 320
Richten Sie IAM-Rollen für Servicekonten zur Abfrage von Metriken ein 323
Schnittstellen-VPC-Endpunkte 326
Erstellen eines Schnittstellen-VPC-Endpunkts für Amazon Managed Service für Prometheus 327
Fehlerbehebung 330
Fehler 429 oder Limit überschritten 330
Ich sehe doppelte Beispiele 332
Ich sehe Fehler bei Beispiel-Zeitstempeln 332
Mir wird eine Fehlermeldung im Zusammenhang mit einem Limit angezeigt 332
Ihre lokale Prometheus-Server-Ausgabe überschreitet das Limit. 333
Einige meiner Daten werden nicht angezeigt 334
HTTP 422-Fehler: Abfrage-Timeout 337
Tagging 338

Markieren von Workspaces ... 339
Einem Workspace ein Tag hinzufügen ... 340
Tags für einen Workspace anzeigen ... 342
Bearbeiten von Tags für einen Workspace ... 343
Ein Tag aus einem Workspace entfernen ... 344
Namespaces von Regelgruppen markieren ... 345
Hinzufügen eines Tags zum Regelgruppen-Namespace ... 346
Anzeigen von Tags für einen Regelgruppen-Namespace ... 348
Bearbeiten von Tags für einen Regelgruppen-Namespace ... 349
Entfernen Sie ein Tag aus einem Regelgruppen-Namespace ... 350
Servicekontingente ... 353
Servicekontingente ... 353
Standardkontingente für aktive Serien ... 362
Skalierung über das Standardkontingent hinaus ... 363
Drosselung der Aufnahme ... 363
Zusätzliche Limits für erfasste Daten ... 365
API-Referenz ... 366
Amazon Managed Service für Prometheus APIs ... 366
Verwenden von Amazon Managed Service für Prometheus mit einem SDK AWS ... 367
Prometheus-compatible APIs ... 367
CreateAlertManagerAlerts ... 368
DeleteAlertManagerSilence ... 369
GetAlertManagerStatus ... 370
GetAlertManagerSilence ... 371
GetLabels ... 373
GetMetricMetadata ... 375
GetSeries ... 376
ListAlerts ... 378
ListAlertManagerAlerts ... 379
ListAlertManagerAlertGroups ... 381
ListAlertManagerReceivers ... 383
ListAlertManagerSilences ... 384
ListRules ... 385
PutAlertManagerSilences ... 388
QueryMetrics ... 390
RemoteWrite ... 392

Dokumentverlauf 394
..... cdii

Was Amazon Managed Service für Prometheus?

Amazon Managed Service für Prometheus ist serverlos, Prometheus-compatible Überwachungsservice für Container-Metriken, der die sichere Überwachung von Container-Umgebungen in großem Maßstab erleichtert. Mit Amazon Managed Service für Prometheus können Sie dasselbe Open-Source-Prometheus-Datenmodell und dieselbe Abfragesprache verwenden, die Sie heute verwenden, um die Leistung Ihrer containerisierten Workloads zu überwachen, und außerdem von verbesserter Skalierbarkeit, Verfügbarkeit und Sicherheit profitieren, ohne die zugrunde liegende Infrastruktur verwalten zu müssen.

Amazon Managed Service für Prometheus skaliert automatisch die Erfassung, Speicherung und Abfrage von Betriebsmetriken, wenn Workloads nach oben oder unten skaliert werden. Es lässt sich integrieren mit AWS Sicherheitsdienste für einen schnellen und sicheren Zugriff auf Daten.

Amazon Managed Service für Prometheus ist so konzipiert, dass es hochverfügbar ist und mehrere Availability Zones verwendet (Multi-AZ) Bereitstellungen. In einen Workspace erfasste Daten werden über drei Availability Zones in derselben Region repliziert.

Amazon Managed Service für Prometheus funktioniert mit Containerclustern, die im Amazon Elastic Kubernetes Service und in selbstverwalteten Kubernetes-Umgebungen ausgeführt werden.

Mit Amazon Managed Service für Prometheus verwenden Sie dasselbe Open-Source-Prometheus-Datenmodell und dieselbe PromQL-Abfragesprache wie Prometheus. Entwicklungsteams können PromQL verwenden, um Metriken zu filtern, zu aggregieren und Alarme zu generieren und sich schnell einen Überblick über die Leistung zu verschaffen, ohne dass Codeänderungen erforderlich sind. Amazon Managed Service für Prometheus bietet flexible Abfragefunktionen ohne Betriebskosten und Komplexität.

In einen Workspace aufgenommene Metriken werden standardmäßig 150 Tage lang gespeichert und dann automatisch gelöscht. Du kannst den Aufbewahrungszeitraum anpassen, indem du deinen Workspace auf maximal 1095 Tage (drei Jahre) konfigurierst. Weitere Informationen findest du unter Konfiguriere deinen Workspace.

Unterstützte Regionen

Amazon Managed Service für Prometheus unterstützt derzeit die folgenden Regionen:

Name der Region	Region	Endpunkt	Protocol (Protokoll)
USA Ost (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 Ost (Nord-Virginia)	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 West	us-west-1	aps.us-west-1.amazonaws.com	HTTPS
		aps-workspaces.us-west-1.amazonaws.com	HTTPS

Name der Region	Region	Endpunkt	Protocol (Protokoll)
(Nordkalifornien)		aps-workspaces-fips.us-west-1.amazonaws.com aps-workspaces-fips.us-west-1.api.aws aps-workspaces.us-west-1.api.aws aps-fips.us-west-1.amazonaws.com aps.us-west-1.api.aws aps-fips.us-west-1.api.aws	HTTPS HTTPS HTTPS HTTPS HTTPS HTTPS
USA West (Oregon)	us-west-2	aps.us-west-2.amazonaws.com aps-workspaces.us-west-2.amazonaws.com aps-workspaces-fips.us-west-2.amazonaws.com aps-workspaces-fips.us-west-2.api.aws aps-workspaces.us-west-2.api.aws aps-fips.us-west-2.amazonaws.com aps.us-west-2.api.aws aps-fips.us-west-2.api.aws	HTTPS HTTPS HTTPS HTTPS HTTPS HTTPS HTTPS HTTPS
Afrika (Kapstadt)	af-south-1	aps.af-south-1.amazonaws.com aps-workspaces.af-south-1.amazonaws.com aps-workspaces.af-south-1.api.aws aps.af-south-1.api.aws	HTTPS HTTPS HTTPS HTTPS

Name der Region	Region	Endpunkt	Protocol (Protokoll)
Asien-Pazifik (Hongkong)	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
Asien-Pazifik (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
Asien-Pazifik (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
Asien-Pazifik (Malaysia)	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

Name der Region	Region	Endpunkt	Protocol (Protokoll)
Asien-Pazifik (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
Asien-Pazifik (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
Asien-Pazifik (Neuseeland)	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
Asien-Pazifik (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

Name der Region	Region	Endpunkt	Protocol (Protokoll)
Asien-Pazifik (Seoul)	ap-northeast-2	aps.ap-northeast-2.amazonaws.com aps-workspaces.ap-northeast-2.amazonaws.com aps-workspaces.ap-northeast-2.api.aws aps.ap-northeast-2.api.aws	HTTPS HTTPS HTTPS HTTPS
Asien-Pazifik (Singapur)	ap-southeast-1	aps.ap-southeast-1.amazonaws.com aps-workspaces.ap-southeast-1.amazonaws.com aps-workspaces.ap-southeast-1.api.aws aps.ap-southeast-1.api.aws	HTTPS HTTPS HTTPS HTTPS
Asien-Pazifik (Sydney)	ap-southeast-2	aps.ap-southeast-2.amazonaws.com aps-workspaces.ap-southeast-2.amazonaws.com aps-workspaces.ap-southeast-2.api.aws aps.ap-southeast-2.api.aws	HTTPS HTTPS HTTPS HTTPS
Asien-Pazifik (Taipeh)	ap-east-2	aps.ap-east-2.amazonaws.com aps-workspaces.ap-east-2.amazonaws.com aps-workspaces.ap-east-2.api.aws aps.ap-east-2.api.aws	HTTPS HTTPS HTTPS HTTPS

Name der Region	Region	Endpunkt	Protocol (Protokoll)	
Asien-Pazifik (Thailand)	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	
Asien-Pazifik (Tokio)	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	
Kanada (Zentral)	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	

Name der Region	Region	Endpunkt	Protocol (Protokoll)
Kanada West (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
Europa (Frankfurt)	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
Europa (Irland)	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

Name der Region	Region	Endpunkt	Protocol (Protokoll)
Europa (London)	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
Europa (Mailand)	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
Europa (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
Europa (Spanien)	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

Name der Region	Region	Endpunkt	Protocol (Protokoll)
Europa (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
Europa (Zürich)	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
Israel (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
Mexiko (Zentral)	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

Name der Region	Region	Endpunkt	Protocol (Protokoll)	
Naher Osten (Bahrain)	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	
Naher Osten (VAE)	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	
Südamerika (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	

Name der Region	Region	Endpunkt	Protocol (Protokoll)
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	
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 für Prometheus umfasst Endpunkte auf der Steuerungsebene (zur Durchführung von Arbeitsbereichsverwaltungsaufgaben) und Endpunkte auf Datenebene (zur Arbeit) Prometheus-compatible Daten in einer Workspace-Instanz). Die Endpunkte der Steuerungsebene beginnen mit `aps.*`, und die Endpunkte der Datenebene beginnen mit `aps-workspaces.*`. Endpunkte, die enden in `.amazonaws.com` unterstützt IPv4 und Endpunkte, die auf enden `.api.aws` unterstützt sowohl IPv4 als auch IPv6.

Preisgestaltung

Es fallen Gebühren für die Erfassung und Speicherung von Metriken an. Die Speichergebühren basieren auf der komprimierten Größe der metrischen Beispiele und Metadaten. Weitere Informationen finden Sie unter Preisgestaltung für Amazon Managed Service für Prometheus.

Sie können verwenden AWS Cost Explorer und AWS Kosten- und Nutzungsberichte zur Überwachung Ihrer Gebühren. Weitere Informationen finden Sie unter Erkunden Sie Ihre Daten mit Cost Explorer und Was sind AWS Kosten- und Nutzungsberichte.

Premium-Support

Wenn Sie ein beliebiges Level des abonnieren AWS Premium-Supportpläne, Ihr Premium-Support gilt für Amazon Managed Service für Prometheus.

Erste Schritte mit Amazon Managed Service für Prometheus

Amazon Managed Service for Prometheus ist ein serverloser, Promethus-kompatibler Service zur Überwachung von Container-Metriken, der es einfach macht, Containerumgebungen in großem Maßstab sicher zu überwachen. Dieser Abschnitt führt Sie durch drei wichtige Bereiche der Verwendung von Amazon Managed Service für Prometheus:

- Einen Workspace erstellen — Erstellen Sie einen Amazon Managed Service for Prometheus Workspace, um Ihre Kennzahlen zu speichern und zu überwachen.
- Metriken aufnehmen — Ihr Workspace ist leer, bis Sie Metriken in Ihren Workspace übertragen haben. Sie können Metriken an Amazon Managed Service for Prometheus senden oder Amazon Managed Service for Prometheus Metriken automatisch auslesen lassen.
- Kennzahlen abfragen — Sobald Sie Metriken als Daten in Ihrem Workspace haben, können Sie die Daten abfragen, um diese Metriken zu untersuchen oder zu überwachen.

Falls Sie noch nicht damit vertraut sind AWS, finden Sie in diesem Abschnitt auch Informationen zur Einrichtung eines AWS-Konto.

Themen

- Einrichten AWS
- Erstellen eines Workspace in Amazon Managed Service für Prometheus
- Erfassen von Prometheus-Metriken in den Workspace
- Abfragen Ihrer Prometheus-Metriken

Einrichten AWS

Melden Sie sich für eine an AWS-Konto

Um loszulegen mit AWS, du brauchst eine AWS-Konto. Es gibt zwei Möglichkeiten, sich anzumelden AWS. Wenn Sie die vollständige Kontrolle über die Kontokonfiguration und den Speicherort Ihrer Inhalte haben möchten oder wenn Sie regulierte Workloads wie HIPAA oder FedRAMP haben, verwenden Sie Melden Sie sich an für AWS (fortgeschritten). Wenn Sie ein neuer Benutzer sind und ein Konto verwenden möchten, das Sie bereits besitzen, um sich anzumelden AWS und erstelle ein AWS Umgebung mit vorkonfigurierten Standardwerten in select AWS-Regionen, verwenden Melde dich an für AWS.

Erstellen eines Workspace in Amazon Managed Service für Prometheus

Ein Workspace ist ein logischer Bereich, der der Speicherung und Abfrage von Prometheus-Metriken gewidmet ist. Ein Workspace unterstützt eine differenzierte Zugriffskontrolle für die Autorisierung seiner Verwaltung, z. B. für das Aktualisieren, Auflisten, Beschreiben und Löschen sowie für die Erfassung und Abfrage von Metriken. Sie können über einen oder mehrere Workspaces in jeder Region in Ihrem Konto verfügen.

Gehen Sie folgendermaßen vor, um einen Workspace einzurichten.

> **Note**
>
> Ausführlichere Informationen zum Erstellen eines Workspace und zu den verfügbaren Optionen finden Sie unter. Erstellen Sie einen Amazon Managed Service für Prometheus Workspace

So erstellen Sie einen Workspace in Amazon Managed Service für Prometheus

1. Öffnen Sie die Amazon Managed Service for Prometheus-Konsole unter. https://console.aws.amazon.com/prometheus/
2. Geben Sie unter Workspace-Alias einen Alias für den neuen Workspace ein.

   Workspace-Aliase sind benutzerfreundliche Namen, anhand derer Sie Ihre Workspaces leichter identifizieren können. Sie müssen nicht einmalig sein. Zwei Arbeitsbereiche könnten denselben Alias haben, aber alle Arbeitsbereiche haben eindeutige Arbeitsbereiche IDs, die von Amazon Managed Service für Prometheus generiert werden.
3. (Optional) Zum Hinzufügen von Tags zum Namespace wählen Sie Neues Tag hinzufügen aus.

   Geben Sie für Key (Schlüssel) einen Namen für das Tag ein. Sie können einen optionalen Wert für das Tag unter Value (Wert) hinzufügen.

   (Optional) Zum Hinzufügen eines weiteren Tags wählen Sie Add new tag (Neues Tag hinzufügen) erneut aus.
4. Wählen Sie Workspace erstellen.

Die Detailseite des Workspace wird angezeigt. Dadurch werden Informationen wie Status, ARN, Workspace-ID und Endpunkt URLs für diesen Workspace sowohl für Remote-Schreibvorgänge als auch für Abfragen angezeigt.

Anfänglich lautet der Status wahrscheinlich CREATING (WIRD ERSTELLT). Warten Sie, bis der Status AKTIV lautet, bevor Sie mit der Einrichtung Ihrer metrischen Erfassung fortfahren.

Notieren Sie sich die für Endpoint — Remote Write URL und Endpoint — Query URL URLs angezeigten. Sie benötigen sie, wenn Sie Ihren Prometheus-Server für Remote-Write-Metriken in diesem Workspace konfigurieren und wenn Sie diese Metriken abfragen.

Erfassen von Prometheus-Metriken in den Workspace

Eine Möglichkeit, Metriken zu erfassen, besteht darin, einen eigenständigen Prometheus-Agenten (eine Prometheus-Instance, die im Agentenmodus läuft) zu verwenden, um Metriken aus Ihrem Cluster zu extrahieren und sie zur Speicherung und Überwachung an Amazon Managed Service für Prometheus weiterzuleiten. In diesem Abschnitt wird erklärt, wie Sie die Erfassung von Metriken in Ihren Workspace in Amazon Managed Service für Prometheus von Amazon EKS einrichten, indem Sie mithilfe des Befehls „Helm“ eine neue Instance des Prometheus-Agents einrichten.

Um in Amazon EKS Metriken wie Kubernetes oder Metriken auf Knotenebene zu generieren, können Sie die Amazon EKS-Community-Add-Ons verwenden. Weitere Informationen finden Sie unter Verfügbare Community-Add-Ons im Amazon EKS-Benutzerhandbuch.

Informationen zu anderen Möglichkeiten, Daten in Amazon Managed Service für Prometheus zu erfassung, einschließlich der Sicherung von Metriken und der Erstellung von Hochverfügbarkeitsmetriken, finden Sie unter Nehmen Sie Metriken in Ihren Amazon Managed Service for Prometheus-Workspace auf.

> **Note**
>
> In einen Workspace aufgenommene Metriken werden standardmäßig 150 Tage lang gespeichert und dann automatisch gelöscht. Du kannst die Aufbewahrungsdauer anpassen, indem du deinen Workspace auf maximal 1095 Tage (drei Jahre) konfigurierst. Weitere Informationen findest du unter Konfiguriere deinen Workspace.

Mit den Anweisungen in diesem Abschnitt können Sie Amazon Managed Service für Prometheus schnell einrichten. Es wird davon ausgegangen, dass Sie bereits einen Workspace erstellt haben. In diesem Abschnitt richten Sie einen neuen Prometheus-Server in einem Amazon EKS-Cluster ein, und der neue Server verwendet eine Standardkonfiguration, um als Agent Metriken an Amazon Managed Service for Prometheus zu senden. Für diese Methode müssen die folgenden Voraussetzungen erfüllt sein:

- Sie benötigen einen Amazon EKS-Cluster, aus dem der neue Prometheus-Server Metriken sammelt.
- In Ihrem Amazon EKS-Cluster muss ein Amazon EBS CSI-Treiber installiert sein (von Helm erforderlich).
- Sie müssen Helm CLI 3.0 oder höher verwenden.
- Sie müssen einen Linux- oder macOS-Computer verwenden, um die Schritte in den folgenden Abschnitten auszuführen.

Schritt 1: Hinzufügen neuer Helm-Chart-Repositorys

Geben Sie die folgenden Befehle ein, um neue Helm-Chart-Repositorys hinzuzufügen. Weitere Informationen zu diesen Befehlen finden Sie unter 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
```

Schritt 2: Erstellen eines Prometheus-Namespace

Geben Sie den folgenden Befehl ein, um einen Prometheus-Namespace für den Prometheus-Server und andere Monitoring-Komponenten zu erstellen. *prometheus-agent-namespace*Ersetzen Sie es durch den Namen, den Sie für diesen Namespace verwenden möchten.

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

Schritt 3: Einrichten von IAM-Rollen für Servicekonten

Für diese Erfassungsmethode müssen Sie IAM-Rollen für Servicekonten im Amazon-EKS-Cluster verwenden, in dem der Prometheus-Agent ausgeführt wird.

Mit IAM-Rollen für Servicekonten können Sie eine IAM-Rolle mit einem Kubernetes-Servicekonto verknüpfen. Dieses Dienstkonto kann dann AWS Berechtigungen für die Container in jedem Pod gewähren, der dieses Dienstkonto verwendet. Weitere Informationen finden Sie unter IAM-Rollen für Servicekonten.

Wenn Sie diese Rollen noch nicht eingerichtet haben, folgen Sie den Anweisungen unter Einrichten von Servicerollen für die Erfassung von Metriken aus Amazon-EKS-Clustern, um die Rollen einzurichten. Die Anweisungen in diesem Abschnitt erfordern die Verwendung von `eksctl`. Weitere Informationen finden Sie unter Erste Schritte mit Amazon Elastic Kubernetes Service – `eksctl`.

> **Note**
>
> Wenn Sie nicht auf EKS oder sind AWS und nur den Zugriffsschlüssel und den geheimen Schlüssel für den Zugriff auf Amazon Managed Service for Prometheus verwenden, können Sie das `EKS-IAM-ROLE` basierte SigV4 nicht verwenden.

Schritt 4: Einrichten des neuen Servers und Beginn der Erfassung von Metriken

Gehen Sie wie folgt vor, um den neuen Prometheus Agenten zu installieren und Messwerte an Ihren Workspace in Amazon Managed Service für Prometheus zu senden.

Um einen neuen Prometheus Agenten zu installieren und Messwerte an Ihren Workspace in Amazon Managed Service für Prometheus zu senden

1. Erstellen Sie mithilfe eines Texteditors eine Datei mit dem Namen `my_prometheus_values_yaml` mit folgenden Inhalten.

   - *`IAM_PROXY_PROMETHEUS_ROLE_ARN`*Ersetzen Sie durch den ARN des amp-iamproxy-ingest-role, in dem Sie es erstellt habenEinrichten von Servicerollen für die Erfassung von Metriken aus Amazon-EKS-Clustern.
   - *`WORKSPACE_ID`*Ersetzen Sie es durch die ID Ihres Amazon Managed Service for Prometheus-Workspace.
   - *`REGION`*Ersetzen Sie es durch die Region Ihres Amazon Managed Service for Prometheus-Workspace.

```
## 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. Geben Sie den folgenden Befehl ein, um den Prometheus-Server zu starten.

   - Ersetzen Sie *prometheus-chart-name* durch Ihren Prometheus-Versionsnamen.
   - Ersetzen Sie es *prometheus-agent-namespace* durch den Namen Ihres Prometheus-Namespaces.

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

Abfragen Ihrer Prometheus-Metriken

Jetzt, da die Metriken in den Workspace erfasst wurden, können Sie sie abfragen. Eine übliche Methode, Ihre Metriken abzufragen, besteht darin, einen Service wie Grafana zu verwenden, um die Metriken abzufragen. In diesem Abschnitt erfahren Sie, wie Sie Amazon Managed Grafana verwenden, um Metriken von Amazon Managed Service für Prometheus abzufragen.

> **Note**
>
> Weitere Möglichkeiten, Ihre Amazon Managed Service for Prometheus-Metriken abzufragen oder den Amazon Managed Service for APIs Prometheus zu verwenden, finden Sie unter. Abfragen Ihrer Prometheus-Metriken

In diesem Abschnitt wird davon ausgegangen, dass Sie bereits einen Workspace erstellt haben und Metriken in diesen aufnehmen.

Sie führen Ihre Abfragen mit der Standard-Prometheus-Abfragesprache PromQL durch. Weitere Informationen zu PromQL und seiner Syntax finden Sie unter Abfragen von Prometheus in der Prometheus-Dokumentation.

Amazon Managed Grafana ist ein vollständig verwalteter Service für Open-Source-Grafana, der die Verbindung zu Open-Source-ISVs und AWS Services für die skalierbare Visualisierung und Analyse Ihrer Datenquellen vereinfacht.

Amazon Managed Service für Prometheus unterstützt die Verwendung von Amazon Managed Grafana zum Abrufen von Abfragemetriken in einem Workspace. In der Konsole von Amazon Managed Grafana können Sie einen Workspace in Amazon Managed Service für Prometheus als Datenquelle hinzufügen, indem Sie Ihre vorhandenen Konten für Amazon Managed Service für Prometheus ermitteln. Amazon Managed Grafana verwaltet die Konfiguration der Authentifizierungsdaten, die für den Zugriff auf Amazon Managed Service für Prometheus erforderlich sind. Detaillierte Anweisungen zum Herstellen einer Verbindung zu Amazon Managed Service für Prometheus von Amazon Managed Grafana aus finden Sie in den Anweisungen im Benutzerhandbuch von Amazon Managed Grafana.

Sie können Ihre Alarme für Amazon Managed Service für Prometheus auch in Amazon Managed Grafana anzeigen. Anweisungen zum Einrichten der Integration mit Benachrichtigungen finden Sie unter Integrieren Sie Benachrichtigungen in Amazon Managed Grafana oder Open-Source-Grafana.

> **Note**
>
> Wenn Sie Ihren Workspace in Amazon Managed Grafana-für die Verwendung einer privaten VPC konfiguriert haben, müssen Sie Ihren Workspace in Amazon Managed Service für Prometheus mit derselben VPC verbinden. Weitere Informationen finden Sie unter Verbindung zu Amazon Managed Grafana in einer privaten VPC herstellen.

Amazon Managed Service für Prometheus-Workspaces verwalten

Ein Workspace ist ein logischer Bereich, der der Speicherung und Abfrage von Prometheus-Metriken gewidmet ist. Ein Workspace unterstützt eine differenzierte Zugriffskontrolle für die Autorisierung seiner Verwaltung, z. B. für das Aktualisieren, Auflisten, Beschreiben und Löschen sowie für die Erfassung und Abfrage von Metriken. Sie können über einen oder mehrere Workspaces in jeder Region in Ihrem Konto verfügen.

Verwenden Sie die Verfahren in diesem Abschnitt, um Ihre Workspaces von Amazon Managed Service für Prometheus zu erstellen und zu verwalten.

Themen

- Erstellen Sie einen Amazon Managed Service für Prometheus Workspace
- Konfiguriere deinen Workspace
- Bearbeiten Sie einen Workspace-Alias
- Finden Sie Ihre Amazon Managed Service for Prometheus Workspace-Details, einschließlich ARN
- Löschen Sie einen Amazon Managed Service for Prometheus-Workspace

Erstellen Sie einen Amazon Managed Service für Prometheus Workspace

Gehen Sie wie folgt vor, um einen Workspace in Amazon Managed Service für Prometheus zu erstellen. Sie können wählen, ob Sie die Amazon Managed Service for Prometheus-Konsole AWS CLI oder die Amazon Managed Service for Prometheus-Konsole verwenden möchten.

> **Note**
>
> Wenn Sie einen Amazon EKS-Cluster ausführen, können Sie mit AWS Controllers for Kubernetes auch einen neuen Workspace erstellen.

Um einen Workspace mit dem zu erstellen AWS CLI

1. Verwenden Sie den folgenden Befehl, um den Workspace zu erstellen. In diesem Beispiel wird ein Workspace mit dem Namen `my-first-workspace` erstellt, Sie können jedoch einen anderen Alias (oder keinen Alias) verwenden, wenn Sie möchten. Workspace-Aliase sind benutzerfreundliche Namen, anhand derer Sie Ihre Workspaces leichter identifizieren können. Sie müssen nicht einmalig sein. Zwei Workspaces können denselben Alias haben, aber alle Workspaces haben eindeutige Workspace-IDs, die von Amazon Managed Service für Prometheus generiert werden.

   (Optional) Um Ihren eigenen KMS-Schlüssel zum Verschlüsseln der in Ihrem Workspace gespeicherten Daten zu verwenden, können Sie den `kmsKeyArn` Parameter zusammen mit dem zu AWS KMS verwendenden Schlüssel verwenden. Amazon Managed Service for Prometheus berechnet Ihnen zwar keine Gebühren für die Verwendung von kundenverwalteten Schlüsseln, es können jedoch Kosten im Zusammenhang mit Schlüsseln von anfallen. AWS Key Management Service Weitere Informationen zur Verschlüsselung von Daten im Workspace mit Amazon Managed Service for Prometheus oder zur Erstellung, Verwaltung und Verwendung Ihres eigenen kundenverwalteten Schlüssels finden Sie unter Verschlüsselung im Ruhezustand.

   Parameter in Klammern (`[]`) sind optional. Schließen Sie die Klammern nicht in Ihrem Befehl ein.

   ```
   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]
   ```

   Dieser Befehl gibt die folgenden Daten zurück:

   - `workspaceId` ist die einmalige ID für diesen Workspace. Notieren Sie sich diese ID.
   - `arn` ist der ARN für diesen Workspace.
   - `status` ist der aktuelle Status des Workspace. Unmittelbar nachdem Sie den Workspace erstellt haben, wird dies wahrscheinlich `CREATING` sein.
   - `kmsKeyArn` ist der vom Kunden verwaltete Schlüssel, der zur Verschlüsselung der Workspace-Daten verwendet wird, falls angegeben.

> **Note**
>
> Workspaces, die mit kundenverwalteten Schlüsseln erstellt wurden, können keine von AWS verwalteten Sammler für die Datenerfassung verwenden.
> Wählen Sie sorgfältig aus, ob Sie vom Kunden verwaltete Schlüssel oder AWS eigene Schlüssel verwenden möchten. Workspaces, die mit vom Kunden verwalteten Schlüsseln erstellt wurden, können später nicht mehr in die Verwendung AWS eigener Schlüssel umgewandelt werden (und umgekehrt).

- `tags` listet die Tags des Workspace auf, falls vorhanden.

2. Wenn Ihr `create-workspace` Befehl den Status von `CREATING` zurückgibt, können Sie den folgenden Befehl eingeben, um festzustellen, wann der Workspace bereit ist. *`my-workspace-id`*Ersetzen Sie es durch den Wert, für `workspaceId` den der `create-workspace` Befehl zurückgegeben hat.

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

   Wenn der Befehl `describe-workspace` für `status` `ACTIVE` zurückgibt, ist der Workspace einsatzbereit.

Erstellen eines Workspace mithilfe der Konsole von Amazon Managed Service für Prometheus

1. Öffnen Sie die Amazon Managed Service for Prometheus-Konsole unter. https://console.aws.amazon.com/prometheus/
2. Wählen Sie Erstellen aus.
3. Geben Sie unter Workspace-Alias einen Alias für den neuen Workspace ein.

   Workspace-Aliase sind benutzerfreundliche Namen, anhand derer Sie Ihre Workspaces leichter identifizieren können. Sie müssen nicht einmalig sein. Zwei Workspaces können denselben Alias haben, aber alle Workspaces haben eindeutige Workspace-IDs, die von Amazon Managed Service für Prometheus generiert werden.
4. (Optional) Um Ihren eigenen KMS-Schlüssel zur Verschlüsselung der in Ihrem Workspace gespeicherten Daten zu verwenden, können Sie die Option Verschlüsselungseinstellungen anpassen auswählen und den zu AWS KMS verwendenden Schlüssel auswählen (oder einen neuen erstellen). Sie können einen Schlüssel in Ihrem Konto aus der Dropdown-Liste auswählen

oder den ARN für einen beliebigen Schlüssel eingeben, auf den Sie Zugriff haben. Amazon Managed Service for Prometheus berechnet Ihnen zwar keine Gebühren für die Verwendung von kundenverwalteten Schlüsseln, es können jedoch Kosten im Zusammenhang mit Schlüsseln von anfallen. AWS Key Management Service

Weitere Informationen zur Verschlüsselung von Daten im Workspace mit Amazon Managed Service for Prometheus oder zur Erstellung, Verwaltung und Verwendung Ihres eigenen kundenverwalteten Schlüssels finden Sie unter Verschlüsselung im Ruhezustand.

> **Note**
>
> Workspaces, die mit vom Kunden verwalteten Schlüsseln erstellt wurden, können keine von AWS verwalteten Sammler für die Datenerfassung verwenden.
> Wählen Sie sorgfältig aus, ob Sie vom Kunden verwaltete Schlüssel oder AWS eigene Schlüssel verwenden möchten. Workspaces, die mit vom Kunden verwalteten Schlüsseln erstellt wurden, können später nicht mehr in die Verwendung AWS eigener Schlüssel umgewandelt werden (und umgekehrt).

5. (Optional) Wählen Sie Neues Tag hinzufügen, um ein oder mehrere Tags zum Workspace hinzuzufügen. Geben Sie dann für Schlüssel einen Namen für das Tag ein. Sie können einen optionalen Wert für das Tag unter Value (Wert) hinzufügen.

   (Optional) Zum Hinzufügen eines weiteren Tags wählen Sie Add new tag (Neues Tag hinzufügen) erneut aus.

6. Wählen Sie Workspace erstellen.

   Die Detailseite des Workspace wird angezeigt. Dadurch werden Informationen wie Status, ARN, Workspace-ID und Endpunkt-URLs für diesen Workspace sowohl für Remote-Write-Vorgänge als auch für Abfragen angezeigt.

   Der Status lautet WIRD ERSTELLT, bis der Workspace bereit ist. Warten Sie, bis der Status AKTIV lautet, bevor Sie mit der Einrichtung Ihrer metrischen Erfassung fortfahren.

   Notieren Sie sich die URLs, die für Endpunkt – Remote-Write-URL und Endpunkt – Query-URL angezeigt werden. Sie benötigen sie, wenn Sie Ihren Prometheus-Server für Remote-Write-Metriken in diesem Workspace konfigurieren und wenn Sie diese Metriken abfragen.

Weitere Informationen zum Erfassen von Metriken im Workspace finden Sie unter Erfassen von Prometheus-Metriken in den Workspace.

Konfiguriere deinen Workspace

Sie können Ihren Workspace für Folgendes konfigurieren:

- Definieren Sie Labelsätze und definieren Sie Grenzwerte für die aktiven Zeitreihen, die Ihren definierten Labelsätzen entsprechen. Ein Labelsatz besteht aus einem oder mehreren Labels, bei denen es sich um name/value Paare handelt, die dazu beitragen, Zeitreihenmetriken einen Kontext zu geben.

  Indem Sie Labelsätze definieren und aktive Zeitreihenlimits festlegen, können Sie Spitzenwerte in einem Mandanten oder einer Quelle so begrenzen, dass sie sich nur auf diesen Mandanten oder diese Quelle auswirken. Wenn Sie beispielsweise ein Limit von 1.000.000 aktiven Zeitreihen für den Labelsatz festlegen und die Anzahl der aufgenommenen Zeitreihen`team=A env=prod`, die diesem Labelsatz entsprechen, den Grenzwert überschreitet, werden nur die Zeitreihen gedrosselt, die dem Labelsatz entsprechen. Auf diese Weise sind andere Mandanten oder Metrikquellen nicht betroffen.

  Weitere Informationen zu Labels in Prometheus finden Sie unter Datenmodell.
- Legen Sie einen Aufbewahrungszeitraum fest, um die Anzahl der Tage zu definieren, für die die Daten im Workspace aufbewahrt werden sollen.
- Legen Sie ein Zeitfenster fest, in dem die Reihenfolge nicht eingehalten wird, damit der Arbeitsbereich Proben annehmen kann, die nicht in chronologischer Reihenfolge eingehen. Alle Arbeitsbereiche haben ein standardmäßiges Zeitfenster von 60 Sekunden, wenn sie nicht in der richtigen Reihenfolge angeordnet sind. Wenn Sie diesen Wert auf 0 setzen, wird die Funktion deaktiviert und Proben, die nicht in der richtigen Reihenfolge sind, werden verworfen.
- Legen Sie einen Offset für Regelabfragen fest, um eine globale Verzögerung vor der Ausführung von Regelauswertungsabfragen einzuführen. Dadurch werden Verzögerungen bei der Aufnahme oder fehlerhafte Stichproben ausgeglichen, sodass die Regeln anhand eines vollständigeren Datensatzes ausgewertet werden. Ein auf Regelgruppenebene konfigurierter Abfrageversatz hat Vorrang vor dieser Einstellung auf Workspace-Ebene. Weitere Informationen finden Sie unter Verwenden Sie Offset in Abfragen, um Verzögerungen bei der Datenerfassung zu beheben.

Um Ihren Workspace zu konfigurieren

1. Öffnen Sie die Amazon Managed Service for Prometheus-Konsole unter. https://console.aws.amazon.com/prometheus/
2. Wählen Sie in der oberen linken Ecke der Seite das Menüsymbol und dann Alle Workspaces.
3. Wählen Sie die Workspace-ID des Workspace aus.
4. Wählen Sie die Registerkarte Workspace-Konfigurationen.
5. Um die Aufbewahrungsfrist für den Workspace festzulegen, wählen Sie im Abschnitt Aufbewahrungszeitraum die Option Bearbeiten aus. Geben Sie dann die neue Aufbewahrungsfrist in Tagen an. Das Maximum beträgt 1095 Tage (drei Jahre).
6. Um das Zeitfenster für nicht in der richtigen Reihenfolge festzulegen, wählen Sie Bearbeiten im Abschnitt Zeitfenster für nicht in der richtigen Reihenfolge. Geben Sie dann das Zeitfenster in Sekunden an. Alle Arbeitsbereiche haben standardmäßig ein Zeitfenster von 60 Sekunden, das nicht in der richtigen Reihenfolge ist. Der Mindestwert ist 0 Sekunden. Wenn er auf 0 gesetzt ist, werden Proben, die nicht in der richtigen Reihenfolge sind, verworfen. Der Höchstwert beträgt 600 Sekunden. Informationen zum Anfordern eines höheren Maximums finden Sie unterServicekontingente von Amazon Managed Service für Prometheus.
7. Um den Offset für die Regelabfrage festzulegen, wählen Sie im Abschnitt Offset für Regelabfragen die Option Bearbeiten aus. Geben Sie dann den Offset in Sekunden an. Der Standardwert für neue Arbeitsbereiche ist 60 Sekunden. Der Mindestwert ist 0 Sekunden und der Höchstwert ist 86.400 Sekunden (24 Stunden).

   > **Note**
   >
   > Bestehende Workspaces haben einen Standardregelabfrage-Offset von 0 Sekunden, den Sie über die Konsole aktualisieren können.
   > Wenn Sie ein Zeitfenster konfiguriert haben, das nicht in der richtigen Reihenfolge ist, sollten Sie erwägen, den Offset für die Regelabfrage auf einen Wert festzulegen, der größer oder gleich dem Zeitfenster ist, in dem das Zeitfenster nicht in der richtigen Reihenfolge ist. Dadurch wird sichergestellt, dass Stichproben, die nicht in der richtigen Reihenfolge sind, in die Regelauswertungen einbezogen werden. Ein Regelabfrage-Offset verzögert Regelauswertungen und alle daraus resultierenden Warnungen um die konfigurierte Dauer.
8. Um Labelsätze und ihre aktiven Serienlimits hinzuzufügen oder zu ändern, wählen Sie im Abschnitt Labelsätze die Option Bearbeiten aus. Führen Sie dann die folgenden Schritte aus:

a. (Optional) Geben Sie im Feld Standard-Bucketlimit einen Wert ein, um ein Limit für die maximale Anzahl aktiver Zeitreihen festzulegen, die in den Workspace aufgenommen werden können. Dabei werden nur Zeitreihen gezählt, die keinem definierten Labelsatz entsprechen.

b. Um einen Labelsatz zu definieren, geben Sie unter Aktives Serienlimit einen Grenzwert für aktive Zeitreihen für den neuen Labelsatz ein.

   Geben Sie dann ein Label und einen Wert für ein Label ein, das im Labelsatz verwendet werden soll, und wählen Sie Label hinzufügen.

c. (Optional) Um einen weiteren Labelsatz zu definieren, wählen Sie „Weiteren Labelsatz hinzufügen“ und wiederholen Sie die vorherigen Schritte.

9. Klicken Sie auf Save changes (Änderungen speichern), wenn Sie fertig sind.

Bearbeiten Sie einen Workspace-Alias

Sie können einen Workspace bearbeiten, um seinen Alias zu ändern. Geben Sie den folgenden Befehl ein, um den Workspace-Alias mithilfe von AWS CLI zu ändern.

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

Einen Workspace mithilfe der Konsole von Amazon Managed Service für Prometheus bearbeiten

1. Öffnen Sie die Amazon Managed Service for Prometheus-Konsole unter. https://console.aws.amazon.com/prometheus/
2. Wählen Sie in der oberen linken Ecke der Seite das Menüsymbol und dann Alle Workspaces.
3. Wählen Sie die Workspace-ID des Workspace aus, den Sie bearbeiten möchten, und wählen Sie dann Bearbeiten aus.
4. Geben Sie einen neuen Alias für den Workspace ein und wählen Sie dann Speichern.

Finden Sie Ihre Amazon Managed Service for Prometheus Workspace-Details, einschließlich ARN

Sie finden die Details Ihres Amazon Managed Service for Prometheus-Workspace entweder über die AWS Konsole oder über. AWS CLI

Console

So finden Sie Ihre Workspace-Details mithilfe der Amazon Managed Service for Prometheus-Konsole

1. Öffnen Sie die Amazon Managed Service for Prometheus-Konsole unter. https://console.aws.amazon.com/prometheus/
2. Wählen Sie in der oberen linken Ecke der Seite das Menüsymbol und dann Alle Workspaces.
3. Wählen Sie die Workspace-ID des Workspace aus. Daraufhin werden Details zu deinem Workspace angezeigt, darunter:

   - Aktueller Status — Der Status deines Workspace, zum Beispiel Aktiv, wird unter Status angezeigt.
   - ARN — Der Workspace-ARN wird unter ARN angezeigt.
   - ID — Die Workspace-ID wird unter Workspace-ID angezeigt.
   - URLs — Die Konsole zeigt mehrere URLs für den Workspace an, einschließlich der URLs zum Schreiben in den Workspace oder zum Abfragen von Daten aus dem Workspace.

   > **Note**
   >
   > Standardmäßig sind die angegebenen URLs die IPv4-URLs. Sie können auch Dualstack-URLs (IPv4 und IPv6 werden unterstützt) verwenden. Diese sind identisch, befinden sich jedoch in der Domäne und nicht in der Standardeinstellung`api.aws. amazonaws.com` Wenn Sie beispielsweise Folgendes sehen würden (eine IPv4-URL):
   >
   > ```
   > https://aps-workspaces.us-east-1.amazonaws.com/workspaces/ws-abcd1234-ef56-7890-ab12-example/api/v1/remote_write
   > ```
   >
   > Sie könnten eine Dual-Stack-URL (einschließlich Unterstützung für IPv6) wie folgt erstellen:
   >
   > ```
   > https://aps-workspaces.us-east-1.api.aws/workspaces/ws-abcd1234-ef56-7890-ab12-example/api/v1/remote_write
   > ```

Unter diesem Abschnitt befinden sich Registerkarten mit Informationen zu Regeln, Alert Manager, Protokollen, Konfiguration und Tags.

AWS CLI

Um die Details zu deinem Workspace zu finden, verwende den AWS CLI

Der folgende Befehl gibt die Details des Workspace zurück. Sie müssen sie *my-workspace-id* durch die Workspace-ID des Workspace ersetzen, für den Sie die Details wünschen.

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

Dadurch werden Details zu deinem Workspace zurückgegeben, darunter:

- Aktueller Status — Der Status deines Workspace wird `ACTIVE` beispielsweise in der `statusCode` Unterkunft angezeigt.
- ARN — Der Workspace-ARN wird in der `arn` Eigenschaft zurückgegeben.
- URLs — AWS CLI Gibt die Basis-URL für den Workspace in der `prometheusEndpoint` Eigenschaft zurück.

> **Note**
>
> Standardmäßig ist die zurückgegebene URL die IPv4-URL. Sie können in der Domain `api.aws` auch eine Dual-Stack-URL (IPv4 und IPv6 unterstützt) anstelle der Standard-URL verwenden. `amazonaws.com` Wenn Sie beispielsweise Folgendes sehen würden (eine IPv4-URL):
>
> ```
> https://aps-workspaces.us-east-1.amazonaws.com/workspaces/ws-abcd1234-ef56-7890-ab12-example/
> ```
>
> Sie könnten eine Dual-Stack-URL (einschließlich Unterstützung für IPv6) wie folgt erstellen:
>
> ```
> https://aps-workspaces.us-east-1.api.aws/workspaces/ws-abcd1234-ef56-7890-ab12-example/
> ```

Sie können auch die Remote-Write- und Abfrage-URLs für den Workspace erstellen, indem Sie jeweils `/api/v1/remote_write` oder `/api/v1/query` hinzufügen.

Löschen Sie einen Amazon Managed Service for Prometheus-Workspace

Beim Löschen eines Workspace werden die Daten gelöscht, die in den Workspace aufgenommen wurden.

> **Note**
>
> Durch das Löschen eines Amazon Managed Service for Prometheus-Workspace werden nicht automatisch alle AWS verwalteten Collectors gelöscht, die Metriken sammeln und an den Workspace senden. Weitere Informationen finden Sie unter Suchen und Löschen von Scrapern.

Um einen Workspace zu löschen, verwenden Sie AWS CLI

Verwenden Sie den folgenden Befehl:

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

Löschen Sie einen Arbeitsbereich über die Konsole von Amazon Managed Service für Prometheus wie folgt

1. Öffnen Sie die Amazon Managed Service for Prometheus-Konsole unter. https://console.aws.amazon.com/prometheus/
2. Wählen Sie in der oberen linken Ecke der Seite das Menüsymbol und dann Alle Workspaces.
3. Wählen Sie die Workspace-ID des Workspace aus, den Sie löschen möchten, und wählen Sie dann Löschen aus.
4. Geben Sie in das Bestätigungsfeld **`delete`** ein und wählen Sie dann Löschen.

Nehmen Sie Metriken in Ihren Amazon Managed Service for Prometheus-Workspace auf

Metriken müssen in Ihren Amazon Managed Service for Prometheus-Workspace aufgenommen werden, bevor Sie diese Metriken abfragen oder Alerts darauf hinweisen können. In diesem Abschnitt wird erklärt, wie Sie die Erfassung von Metriken in Ihren Workspace einrichten können.

> **Note**
>
> In einen Workspace aufgenommene Metriken werden standardmäßig 150 Tage lang gespeichert und dann automatisch gelöscht. Du kannst die Aufbewahrungsdauer anpassen, indem du deinen Workspace auf maximal 1095 Tage (drei Jahre) konfigurierst. Weitere Informationen findest du unter Konfiguriere deinen Workspace.

Es gibt zwei Methoden, um Metriken in Ihren Workspace in Amazon Managed Service für Prometheus zu erfassen.

- Verwendung eines AWS verwalteten Collectors — Amazon Managed Service for Prometheus bietet einen vollständig verwalteten, agentenlosen Scraper zum automatischen Scrapen von Metriken aus Ihren Amazon Elastic Kubernetes Service (Amazon EKS) -Clustern. Scraping ruft die Metriken automatisch von Prometheus-kompatiblen Endpunkten ab.
- Verwendung eines vom Kunden verwalteten Kollektors – Sie haben viele Möglichkeiten, Ihren eigenen Kollektor zu verwalten. Zwei der am häufigsten verwendeten Collectors sind die Installation Ihrer eigenen Instanz von Prometheus, die Ausführung im Agentenmodus oder die Verwendung von AWS Distro for. OpenTelemetry Diese beiden werden in den folgenden Abschnitten ausführlich beschrieben.

  Kollektoren senden Metriken an Amazon Managed Service für Prometheus, indem die Remote-Write-Feature von Prometheus verwenden. Sie können Metriken direkt an Amazon Managed Service für Prometheus senden, indem Sie die Prometheus-Remote-Write-Feature in Ihrer eigenen Anwendung verwenden. Weitere Informationen zur direkten Verwendung von Remote-Write und Remote Write-Konfigurationen finden Sie unter remote_write in der Prometheus-Dokumentation.

Themen

- Metriken aufnehmen mit AWS verwaltete Sammler

- Kundenverwaltete Kollektoren

Metriken aufnehmen mit AWS verwaltete Sammler

Ein häufiger Anwendungsfall für Amazon Managed Service für Prometheus ist die Überwachung von Kubernetes-Clustern, die von Amazon Elastic Kubernetes Service (Amazon EKS) verwaltet werden. Kubernetes-Cluster und viele Anwendungen, die in Amazon EKS ausgeführt werden, exportieren ihre Metriken automatisch für Prometheus-compatible Scraper zum Zugreifen.

> **Note**
>
> Amazon EKS stellt API-Server-Metriken zur Verfügung, `kube-controller-manager` Metriken und `kube-scheduler` Metriken in einem Cluster. Viele andere Technologien und Anwendungen, die in Kubernetes-Umgebungen ausgeführt werden, bieten Prometheus-compatible Metriken. Eine vollständige Liste der verfügbaren Exporter finden Sie unter Exporter und Integrationen in der Prometheus-Dokumentation.

Amazon Managed Service für Prometheus bietet einen vollständig verwalteten Scraper ohne Agenten oder Kollektor, der automatisch entdeckt und abruft Prometheus-compatible Metriken. Sie müssen Agenten oder Scraper nicht verwalten, installieren, patchen oder warten. Ein Kollektor von Amazon Managed Service für Prometheus bietet eine zuverlässige, stabile, hochverfügbare und automatisch skalierte Erfassung von Metriken für Ihren Amazon-EKS-Cluster. Die von Amazon Managed Service for Prometheus verwalteten Collectors funktionieren mit Amazon EKS-Clustern, einschließlich EC2 und Fargate.

Ein Sammler von Amazon Managed Service for Prometheus erstellt eine Elastic-Network-Schnittstelle (ENI) pro Subnetz, die bei der Erstellung des Scrapers angegeben wurde. Der Sammler durchsucht die Metriken über diese ENIs und leitet die Daten mithilfe von `remote_write` eines VPC-Endpunkts an Ihren Workspace in Amazon Managed Service for Prometheus weiter. Diese erhobenen Daten werden niemals über das öffentliche Internet übertragen.

Die folgenden Themen enthalten weitere Informationen zur Verwendung eines Kollektors von Amazon Managed Service für Prometheus in Ihrem Amazon-EKS-Cluster und zu den gesammelten Metriken.

Themen

- Richten Sie verwaltete Collectors für Amazon EKS ein

- Richten Sie verwaltete Prometheus-Sammler für Amazon MSK ein
- Richten Sie verwaltete Prometheus-Collectors für Amazon ein OpenSearch Bedienung
- Was sind Prometheus-compatible Metriken?
- Überwachen Sie Sammler mit verkauften Baumstämmen

Richten Sie verwaltete Collectors für Amazon EKS ein

Um einen Amazon Managed Service for Prometheus Collector zu verwenden, erstellen Sie einen Scraper, der Metriken in Ihrem Amazon EKS-Cluster erkennt und abruft. Sie können auch einen Scraper erstellen, der in Amazon Managed Streaming für Apache Kafka oder in Amazon integriert ist OpenSearch Bedienung. Weitere Informationen finden Sie unter Integrieren Sie Amazon MSK und Integrieren Sie Amazon OpenSearch Service.

- Sie können einen Scraper als Teil Ihrer Amazon-EKS-Cluster-Erstellung erstellen. Weitere Informationen zur Erstellung eines Amazon-EKS-Clusters, einschließlich der Erstellung eines Scrapers, finden Sie unter Erstellen eines Amazon-EKS-Clusters im Amazon-EKS-Benutzerhandbuch.
- Sie können Ihren eigenen Scraper programmgesteuert mit dem erstellen AWS API oder mithilfe der AWS CLI.

Ein Amazon Managed Service für Prometheus-Sammler entfernt Kennzahlen, die Prometheus-compatible. Weitere Informationen zu Prometheus-kompatiblen Metriken finden Sie unter Was sind Prometheus-compatible Metriken?. Amazon EKS-Cluster stellen Metriken für den API-Server bereit. Amazon EKS-Cluster, die eine Kubernetes-Version haben `1.28` oder höher stellen auch Metriken für die zur Verfügung `kube-scheduler` und `kube-controller-manager`. Weitere Informationen finden Sie unter Ruft Rohmetriken der Steuerungsebene im Prometheus-Format ab im Amazon EKS-Benutzerhandbuch.

> **Note**
>
> Beim Scraping von Metriken aus einem Cluster können Gebühren für die Netzwerknutzung anfallen. Eine Möglichkeit, diese Kosten zu optimieren, ist die Konfiguration Ihres `/metrics` Endpunkt, um die bereitgestellten Metriken zu komprimieren (z. B. mit gzip), wodurch die Anzahl der Daten, die über das Netzwerk übertragen werden müssen, reduziert wird. Wie

das geht, hängt von der Anwendung oder Bibliothek ab, die die Metriken bereitstellt. Einige Bibliotheken sind standardmäßig gzip.

In den folgenden Themen wird beschrieben, wie Sie Scraper erstellen, verwalten und konfigurieren.

Themen

- Einen Scraper erstellen
- Ihren Amazon-EKS-Cluster konfigurieren
- Suchen und Löschen von Scrapern
- Scraper-Konfiguration
- Fehlerbehebung der Scraper-Konfiguration
- Scraper-Einschränkungen

Einen Scraper erstellen

Ein Kollektor von Amazon Managed Service für Prometheus besteht aus einem Scraper, der Metriken aus einem Amazon-EKS-Cluster erkennt und sammelt. Amazon Managed Service für Prometheus verwaltet den Scraper für Sie und bietet Ihnen so die Skalierbarkeit, Sicherheit und Zuverlässigkeit, die Sie benötigen, ohne Instances, Agenten oder Scraper selbst verwalten zu müssen.

Es gibt drei Möglichkeiten, einen Scraper zu erstellen:

- Ein Scraper wird automatisch für dich erstellt, wenn du erstellen Sie einen Amazon EKS-Cluster über die Amazon EKS-Konsole und wählen Sie, ob Sie Prometheus-Metriken aktivieren möchten.
- Sie können von der Amazon EKS-Konsole aus einen Scraper für einen vorhandenen Cluster erstellen. Öffnen Sie den Cluster in der Amazon EKS-Konsole, dann, auf der Beobachtbarkeit Tab, wählen Sie Scraper hinzufügen.

  Weitere Informationen zu den verfügbaren Einstellungen finden Sie unter Schalten Sie die Prometheus-Metriken ein in der Amazon EKS-Benutzerhandbuch.
- Sie können einen Scraper erstellen, indem Sie entweder AWS API oder die AWS CLI.

  Diese Optionen werden im folgenden Verfahren beschrieben.

Es gibt einige Voraussetzungen, um Ihren eigenen Scraper zu erstellen:

- Sie haben einen Amazon-EKS-Cluster erstellt.
- In Ihrem Amazon-EKS-Cluster muss die Cluster-Endpunkt-Zugriffskontrolle so eingestellt sein, dass sie privaten Zugriff beinhaltet. Er kann private und öffentliche Daten beinhalten, muss aber mindestens private Daten beinhalten.
- Die Amazon VPC, in der sich der Amazon EKS-Cluster befindet, muss DNS aktiviert.

> **Note**
>
> Der Cluster wird dem Scraper anhand seines Amazon-Ressourcennamens (ARN) zugeordnet. Wenn Sie einen Cluster löschen und dann einen neuen mit demselben Namen erstellen, wird der ARN für den neuen Cluster wiederverwendet. Aus diesem Grund versucht der Scraper, Metriken für den neuen Cluster zu sammeln. Du löscht Scraper getrennt vom Löschen des Clusters.

AWS API

So erstellen Sie einen Scraper unter Verwendung der AWS -API

Verwenden Sie den `CreateScraper` API-Operation, um einen Scraper mit dem zu erstellen AWS API. Im folgenden Beispiel wird ein Scraper in der `us-west-2` Region erstellt. Sie müssen die Informationen zu AWS-Konto, Workspace, Sicherheit und Amazon-EKS-Cluster durch Ihre eigenen IDs ersetzen und die Konfiguration angeben, die für Ihren Scraper verwendet werden soll.

> **Note**
>
> Die Sicherheitsgruppe und die Subnetze sollten auf die Sicherheitsgruppe und die Subnetze für den Cluster festgelegt werden, mit dem Sie eine Verbindung herstellen. Sie müssen mindestens zwei Subnetze in mindestens zwei Availability Zones einfügen.

Das `scrapeConfiguration` ist eine Prometheus-Konfigurations-YAML-Datei, die Base64-codiert ist. Sie können eine allgemeine Zweckkonfiguration mit der `GetDefaultScraperConfiguration`API-Operation herunterladen. Weitere Informationen zum Format des `scrapeConfiguration`, siehe Scraper-Konfiguration.

```
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

So erstellen Sie einen Scraper unter Verwendung der AWS CLI

Benutze das `create-scraper` Befehl zum Erstellen eines Schabers mit dem AWS CLI. Im folgenden Beispiel wird ein Scraper in der `us-west-2` Region erstellt. Sie müssen die Informationen zu AWS-Konto, Workspace, Sicherheit und Amazon-EKS-Cluster durch Ihre eigenen IDs ersetzen und die Konfiguration angeben, die für Ihren Scraper verwendet werden soll.

> **Note**
>
> Die Sicherheitsgruppe und die Subnetze sollten auf die Sicherheitsgruppe und die Subnetze für den Cluster festgelegt werden, zu dem Sie eine Verbindung herstellen. Sie müssen mindestens zwei Subnetze in mindestens zwei Availability Zones einfügen.

Das `scrape-configuration` ist eine Prometheus-Konfigurations-YAML-Datei, die Base64-codiert ist. Sie können eine Allzweckkonfiguration mit dem herunterladen `get-default-scraper-configuration` Befehl. Für weitere Informationen über das Format des `scrape-configuration`, siehe Scraper-Konfiguration.

```
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'}"
```

Im Folgenden finden Sie eine vollständige Liste der Scraper-Operationen, die Sie mit der AWS API verwenden können:

- Erstellen Sie einen Scraper mit der CreateScraper API-Operation.
- Listen Sie Ihre vorhandenen Scraper mit der ListScrapers API-Operation auf.
- Aktualisieren Sie den Alias, die Konfiguration oder das Ziel eines Scrapers mit dem UpdateScraper API-Betrieb.
- Löschen Sie einen Scraper mit der DeleteScraper API-Operation.
- Erfahren Sie mehr über einen Scraper mit der DescribeScraper API-Operation.
- Holen Sie sich eine allgemeine Zweckkonfiguration mit der GetDefaultScraperConfigurationAPI-Operation.

> **Note**
>
> Der Amazon-EKS-Cluster, den Sie scrapen, muss so konfiguriert sein, dass Amazon Managed Service für Prometheus auf die Metriken zugreifen kann. Im nächsten Thema wird beschrieben, wie Sie Ihren Cluster konfigurieren.

Cross-account Einrichten

Gehen Sie wie folgt vor, um einen kontoübergreifenden Scraper zu erstellen, wenn sich Ihr Amazon EKS-Cluster und Amazon Managed Service for Prometheus Workspace in unterschiedlichen Konten

befinden. Sie haben beispielsweise ein Quellkonto `account_id_source` enthält den Amazon EKS-Cluster und ein Zielkonto `account_id_target` enthält den Amazon Managed Service für Prometheus Workspace.

Um einen Scraper in einer kontoübergreifenden Konfiguration zu erstellen

1. Erstelle im Quellkonto eine Rolle `arn:aws:iam::`*`account_id`*`_source:role/Source` und fügen Sie die folgende Vertrauensrichtlinie hinzu.

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

2. Für jede Kombination aus Quelle (Amazon EKS-Cluster) und Ziel (Amazon Managed Service for Prometheus Workspace) müssen Sie eine Rolle erstellen `arn:aws:iam::`*`account_id`*`_target:role/Target` und fügen Sie die folgende Vertrauensrichtlinie mit Berechtigungen für hinzu AmazonPrometheusRemoteWriteAccess.

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

```
        "sts:ExternalId": "scraper_ARN"
      }
  }
}
```

3. Erstellen Sie einen Schaber mit dem `--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. Bestätigen Sie die Erstellung des Scrapers.

```
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"
                }
            }
        }
    ]
}
```

Wechselt zwischen RoleConfiguration und dienstbezogene Rolle

Wenn Sie zurück zu einer dienstverknüpften Rolle wechseln möchten, anstatt zu der `RoleConfiguration` um in einen Amazon Managed Service for Prometheus-Workspace zu schreiben, müssen Sie den `UpdateScraper` und stelle einen Workspace im gleichen Konto wie der Scraper bereit, ohne den `RoleConfiguration`. Das `RoleConfiguration` wird aus dem Scraper entfernt und die dienstgebundene Rolle wird verwendet.

Wenn Sie die Arbeitsbereiche im selben Konto wie der Scraper wechseln und den weiterhin verwenden möchten `RoleConfiguration`, müssen Sie erneut das angeben `RoleConfiguration` auf `UpdateScraper`.

Erstellen eines Scrapers für Arbeitsbereiche, die mit vom Kunden verwalteten Schlüsseln aktiviert sind

So erstellen Sie einen Scraper für die Aufnahme von Metriken in einen Amazon Managed Service for Prometheus-Workspace mit vom Kunden verwaltete Schlüssel, benutze den `--role-configuration` wobei sowohl die Quelle als auch das Ziel auf dasselbe Konto festgelegt sind.

```
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"}'
```

Häufige Fehler beim Erstellen von Scrapern

Im Folgenden sind die häufigsten Probleme aufgeführt, wenn versucht wird, einen neuen Scraper zu erstellen.

- Erforderlich AWS Ressourcen existieren nicht. Die Sicherheitsgruppe, Subnetze, und Amazon EKS-Cluster Der angegebene muss existieren.
- Ungenügender IP-Adressraum. In jedem Subnetz, das Sie an das übergeben, muss mindestens eine IP-Adresse verfügbar sein `CreateScraper` API.

Ihren Amazon-EKS-Cluster konfigurieren

Ihr Amazon-EKS-Cluster muss so konfiguriert sein, dass der Scraper auf Metriken zugreifen kann. Für diese Konfiguration gibt es zwei Optionen:

- Verwenden Sie Amazon EKS auf Einträge zugreifen um Amazon Managed Service für Prometheus-Collectors automatisch Zugriff auf Ihren Cluster zu gewähren.
- Konfigurieren Sie Ihren Amazon EKS-Cluster manuell für verwaltetes Metrik-Scraping.

In den folgenden Themen wird jedes dieser Themen ausführlicher beschrieben.

Konfigurieren Sie Amazon EKS für den Scraper-Zugriff mit Zugriffseinträgen

Die Verwendung von Zugriffseinträgen für Amazon EKS ist der einfachste Weg, Amazon Managed Service for Prometheus Zugriff auf Scrape-Metriken aus Ihrem Cluster zu gewähren.

Der Amazon EKS-Cluster, den Sie scrapen, muss so konfiguriert sein, dass er die API-Authentifizierung zulässt. Der Cluster-Authentifizierungsmodus muss auf einen der folgenden Werte eingestellt werden `API` oder `API_AND_CONFIG_MAP`. Dies ist in der Amazon EKS-Konsole auf der Auf die Konfiguration zugreifen Registerkarte der Cluster-Details. Weitere Informationen finden Sie unter Ermöglichen Sie IAM-Rollen oder Benutzern den Zugriff auf das Kubernetes-Objekt in Ihrem Amazon EKS-Cluster in der Amazon EKS-Benutzerhandbuch.

Sie können den Scraper beim Erstellen des Clusters oder nach dem Erstellen des Clusters erstellen:

- Beim Erstellen eines Clusters — Sie können diesen Zugriff konfigurieren, wenn Sie erstellen Sie einen Amazon EKS-Cluster über die Amazon EKS-Konsole (folgen Sie den Anweisungen, um einen Scraper als Teil des Clusters zu erstellen). Daraufhin wird automatisch eine Zugriffsrichtlinie erstellt, die Amazon Managed Service for Prometheus Zugriff auf die Cluster-Metriken gewährt.
- Hinzufügen, nachdem ein Cluster erstellt wurde — Wenn Ihr Amazon EKS-Cluster bereits existiert, stellen Sie den Authentifizierungsmodus auf eine der folgenden Optionen ein `API` oder `API_AND_CONFIG_MAP`, und alle Scraper, die du erstellst über die Amazon Managed Service für Prometheus-API oder CLI oder über die Amazon EKS-Konsole wird automatisch die richtige Zugriffsrichtlinie für Sie erstellt, und die Scraper haben Zugriff auf Ihren Cluster.

Zugriffsrichtlinie wurde erstellt

Wenn Sie einen Scraper erstellen und Amazon Managed Service for Prometheus eine Zugriffsrichtlinie für Sie erstellen lassen, wird die folgende Richtlinie generiert. Weitere Informationen zu Zugriffseinträgen finden Sie unter Erlauben Sie IAM-Rollen oder Benutzern den Zugriff auf Kubernetes in der Amazon EKS-Benutzerhandbuch.

```
{
    "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"
            ]
        }
    ]
}
```

Manuelles Konfigurieren von Amazon EKS für den Scraper-Zugriff

Wenn Sie es vorziehen, das zu verwenden `aws-auth ConfigMap` Um den Zugriff auf Ihren Kubernetes-Cluster zu kontrollieren, können Sie Amazon Managed Service for Prometheus Scraper weiterhin Zugriff auf Ihre Metriken gewähren. Mit den folgenden Schritten erhält Amazon Managed Service for Prometheus Zugriff auf Scrape-Metriken aus Ihrem Amazon EKS-Cluster.

> **Note**
>
> Für weitere Informationen über `ConfigMap` und auf Einträge zugreifen, siehe Erlauben Sie IAM-Rollen oder Benutzern den Zugriff auf Kubernetes in der Amazon EKS-Benutzerhandbuch.

Dieses Verfahren verwendet `kubectl` und der AWS CLIP. Informationen zum Installieren von `kubectl` finden Sie unter Installieren von kubectl im Amazon-EKS-Benutzerhandbuch.

Um Ihren Amazon EKS-Cluster manuell für verwaltetes Metrik-Scraping zu konfigurieren

1. Erstellen Sie eine Datei mit dem Namen `clusterrole-binding.yml` und dem folgenden Text:

```
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. Führen Sie den folgenden Befehl in Ihrem Cluster aus:

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

Dadurch werden die Cluster-Rollenbindung und die Regel erstellt. In diesem Beispiel wird `aps-collector-role` als Rollenname und `aps-collector-user` als Benutzername verwendet.

3. Der folgende Befehl gibt Ihnen Informationen über den Scraper mit der ID *`scraper-id`*. Dies ist der Scraper, den Sie mit dem Befehl im vorherigen Abschnitt erstellt haben.

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

4. Suchen Sie in den Ergebnissen von `describe-scraper` den `roleArn`. Dieser hat das folgende Format:

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

Amazon EKS benötigt ein anderes Format für diesen ARN. Sie müssen das Format des zurückgegebenen ARN anpassen, damit es im nächsten Schritt verwendet werden kann. Bearbeiten Sie es so, dass es diesem Format entspricht:

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

Zum Beispiel diese ARN:

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

Muss wie folgt umgeschrieben werden:

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

5. Führen Sie den folgenden Befehl in Ihrem Cluster aus, indem Sie den überarbeiteten `roleArn` aus dem vorherigen Schritt sowie Ihren Clusternamen und Ihre Region verwenden:

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

Dadurch kann der Scraper mit der Rolle und dem Benutzer, die Sie in der `clusterrole-binding.yml` Datei erstellt haben, auf den Cluster zugreifen.

Suchen und Löschen von Scrapern

Sie können das verwenden AWS API oder die AWS CLI um die Scraper in deinem Konto aufzulisten oder zu löschen.

> **Note**
>
> Stellen Sie sicher, dass Sie die neueste Version von verwenden AWS CLI oder SDK. Die neueste Version bietet Ihnen die neuesten Features und Funktionen sowie Sicherheitsupdates. Verwenden Sie alternativ AWS CloudShell, das automatisch ein stets aktuelles Befehlszeilenerlebnis bietet.

Die ListScrapers-API-Operation listet alle Scraper in Ihrem Konto auf.

Alternativ können Sie mit dem AWS CLI, rufen Sie an:

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

`ListScrapers` gibt alle Scraper in Ihrem Konto zurück, zum Beispiel:

```
{
    "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"
                }
            }
        }
    ]
}
```

Suchen Sie zum Löschen eines Scrapers mithilfe der `ListScrapers`-Operation die `scraperId` für den Scraper, den Sie löschen möchten, und verwenden Sie dann die DeleteScraper-Operation, um ihn zu löschen.

Alternativ mit dem AWS CLI, rufen Sie an:

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

Scraper-Konfiguration

Mit einem können Sie steuern, wie Ihr Scraper Metriken entdeckt und sammelt Prometheus-compatible Scraper-Konfiguration. Sie können beispielsweise das Intervall ändern, in dem Metriken an den Workspace gesendet werden. Sie können Umetikettierung auch verwenden, um die Etiketten einer Metrik dynamisch neu zu schreiben. Die Scraper-Konfiguration ist eine YAML-Datei, die Teil der Definition des Scrapers ist.

Wenn ein neuer Scraper erstellt wird, geben Sie eine Konfiguration an, indem Sie im API-Aufruf eine Base64-codierte YAML-Datei angeben. Sie können eine allgemeine Zweckkonfigurationsdatei mit dem `GetDefaultScraperConfiguration` Vorgang in der API für Amazon Managed Service für Prometheus herunterladen.

Um die Konfiguration eines Scrapers zu ändern, können Sie den `UpdateScraper` Betrieb. Wenn Sie die Quelle der Metriken aktualisieren müssen (z. B. auf einen anderen Amazon EKS-Cluster), müssen Sie den Scraper löschen und ihn mit der neuen Quelle neu erstellen.

Unterstützte Konfiguration

Informationen zum Scraper-Konfigurationsformat, einschließlich einer detaillierten Aufschlüsselung der möglichen Werte, finden Sie unter Konfiguration in der Prometheus-Dokumentation. Die globalen Konfigurationsoptionen und `<scrape_config>` Optionen beschreiben die am häufigsten benötigten Optionen.

Weil Amazon EKS der einzige unterstützte Service ist, die einzige Service Discovery-Konfiguration (`<*_sd_config>`) unterstützt wird `<kubernetes_sd_config>`.

Die vollständige Liste der erlaubten Konfigurationsabschnitte:

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

Die Einschränkungen in diesen Abschnitten sind nach der Beispielkonfigurationsdatei aufgeführt.

Beispielkonfigurationsdatei

Im Folgenden finden Sie ein Beispiel für eine YAML-Konfigurationsdatei mit einem Scrape-Intervall von 30 Sekunden. Dieses Beispiel beinhaltet Unterstützung für die Kube-API-Server-Metriken sowie die Kube-Controller-Manager- und Kube-Scheduler-Metriken. Weitere Informationen finden Sie unter Ruft Rohmetriken der Steuerungsebene im Prometheus-Format ab im Amazon EKS-Benutzerhandbuch.

```
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
```

Die folgenden Einschränkungen sind spezifisch für AWS verwaltete Sammler:

- Scrape-Intervall – In der Scraper-Konfiguration kann kein Scrape-Intervall von weniger als 30 Sekunden angegeben werden.
- Ziele – Ziele in der `static_config` müssen als IP-Adressen angegeben werden.
- DNS-Auflösung — In Bezug auf den Zielnamen ist der einzige Servername, der in dieser Konfiguration erkannt wird, der Kubernetes-API-Server. `kubernetes.default.svc`. Alle anderen Computernamen müssen anhand der IP-Adresse angegeben werden.
- Autorisierung — Auslassen, wenn keine Autorisierung erforderlich ist. Wenn sie benötigt wird, muss die Autorisierung wie folgt lauten `Bearer`, und muss auf die Datei verweisen `/var/run/secrets/kubernetes.io/serviceaccount/token`. Mit anderen Worten, wenn er verwendet wird, muss der Autorisierungsbereich wie folgt aussehen:

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

  > **Note**
  >
  > `type: Bearer` ist die Standardeinstellung und kann daher weggelassen werden.

Fehlerbehebung der Scraper-Konfiguration

Kollektoren von Amazon Managed Service für Prometheus entdecken und scrapen automatisch Metriken. Aber wie können Sie Fehler beheben, wenn Sie in Ihrem Workspace in Amazon Managed Service für Prometheus eine Metrik nicht sehen, die Sie erwarten würden?

> **Important**
>
> Stellen Sie sicher, dass der private Zugriff für Ihren Amazon EKS-Cluster aktiviert ist. Weitere Informationen finden Sie unter Privater Cluster-Endpunkt in der Amazon EKS-Benutzerhandbuch.

Die up Metrik ist ein hilfreiches Tool. Für jeden Endpunkt, den ein Kollektor von Amazon Managed Service für Prometheus erkennt, verkauft er diese Metrik automatisch. Es gibt drei Status dieser Metrik, die Ihnen helfen können, Fehler im Kollektor zu beheben.

- up ist nicht vorhanden – Wenn für einen Endpunkt keine up-Metrik vorhanden ist, bedeutet das, dass der Kollektor den Endpunkt nicht finden konnte.

  Wenn Sie sicher sind, dass der Endpunkt existiert, gibt es mehrere Gründe, warum der Collector ihn möglicherweise nicht finden kann.

  - Möglicherweise müssen Sie die Scrape-Konfiguration anpassen. Die Entdeckung `relabel_config` muss möglicherweise angepasst werden.
  - Es könnte ein Problem mit dem geben `role` zur Entdeckung verwendet.
  - Die vom Amazon EKS-Cluster verwendete Amazon VPC hat möglicherweise nicht DNS aktiviert, was den Collector daran hindern würde, den Endpunkt zu finden.
- up ist vorhanden, ist aber immer 0 — Wenn up ist vorhanden, aber 0, dann kann der Collector den Endpunkt ermitteln, aber keinen finden Prometheus-compatible Metriken.

  In diesem Fall könnten Sie versuchen, einen `curl` Befehl direkt für den Endpunkt zu verwenden. Sie können überprüfen, ob die Angaben korrekt sind, z. B. das Protokoll (`http` oder `https`), den Endpunkt oder Port, den Sie verwenden. Sie können auch überprüfen, ob der Endpunkt mit einem gültigen `200` antwortet und folgt dem Prometheus-Format. Schließlich darf der Hauptteil der Antwort nicht größer als die maximal zulässige Größe sein. (Für Grenzwerte von AWS verwaltete Collectors finden Sie im folgenden Abschnitt.)
- up ist vorhanden und größer als 0 – Wenn up vorhanden und größer als 0 ist, werden Metriken an Amazon Managed Service für Prometheus gesendet.

  Stellen Sie sicher, dass Sie in Amazon Managed Service für Prometheus (oder Ihrem alternativen Dashboard, z. B. Amazon Managed Grafana) nach den richtigen Metriken suchen. Sie können curl erneut verwenden, um nach erwarteten Daten in Ihrem `/metrics` Endpunkt zu suchen. Vergewissern Sie sich auch, dass Sie andere Grenzwerte nicht überschritten haben, z. B. die

Anzahl der Endpunkte pro Scraper. Sie können die Anzahl der Metrik-Endpunkte überprüfen, die gescrapt werden, indem Sie die Anzahl von überprüfen `up` Metriken, verwenden `count(up)`.

Scraper-Einschränkungen

Die vollständig verwalteten Scraper, die von Amazon Managed Service für Prometheus bereitgestellt werden, unterliegen nur wenigen Einschränkungen.

- Region – Ihr EKS-Cluster, Ihr Managed Scraper und Ihr Workspace in Amazon Managed Service für Prometheus müssen sich alle in derselben AWS -Region befinden.
- Kollektoren – Sie können pro Region und Konto maximal 10 Scraper von Amazon Managed Service für Prometheus haben.

> **Note**
>
> Sie können eine Erhöhung dieses Limits anfordern, indem Sie eine Erhöhung der Quote anfordern.

- Antwort auf Metriken – Der Hauptteil einer Antwort auf eine einzelne `/metrics`-Endpunktanfrage darf nicht mehr als 50 Megabyte (MB) umfassen.
- Endpunkte pro Scraper – Ein Scraper kann maximal 30.000 Endpunkte scrapen.
- Scrape-Intervall – In der Scraper-Konfiguration kann kein Scrape-Intervall von weniger als 30 Sekunden angegeben werden.

Richten Sie verwaltete Prometheus-Sammler für Amazon MSK ein

Um einen Amazon Managed Service for Prometheus Collector zu verwenden, erstellen Sie einen Scraper, der Metriken in Ihrem Amazon Managed Streaming for Apache Kafka-Cluster erkennt und abruft. Sie können auch einen Scraper erstellen, der in Amazon Elastic Kubernetes Service oder Amazon integriert werden kann OpenSearch Dienst. Weitere Informationen finden Sie unter Integrieren Sie Amazon EKS und Integrieren Sie Amazon OpenSearch Service.

Anweisungen zur Erfassung von Verzögerungen bei Verbrauchergruppen, zugesagten Kompensationen und zur Mitgliedschaft bei einem von Kafka verwalteten Exporteur finden Sie unter Erfassen Sie Kennzahlen zu Amazon MSK-Verbrauchergruppen. Dieses Verfahren verwendet AWS Identity and Access Management (IAM) -Authentifizierung und erfordert kein Open Monitoring. Das folgende Verfahren erfasst JMX- und Node Exporter-Metriken aus Open Monitoring.

Einen Scraper erstellen

Ein Amazon Managed Service for Prometheus Collector besteht aus einem Scraper, der Metriken aus einem Amazon MSK-Cluster erkennt und sammelt. Amazon Managed Service für Prometheus verwaltet den Scraper für Sie und bietet Ihnen so die Skalierbarkeit, Sicherheit und Zuverlässigkeit, die Sie benötigen, ohne Instances, Agenten oder Scraper selbst verwalten zu müssen.

Sie können einen Scraper erstellen, indem Sie entweder AWS API oder die AWS CLI wie in den folgenden Verfahren beschrieben.

Es gibt einige Voraussetzungen, um Ihren eigenen Scraper zu erstellen:

- Sie müssen einen Amazon MSK-Cluster erstellt haben.
- Konfigurieren Sie die Sicherheitsgruppe Ihres Amazon MSK-Clusters so, dass eingehender Datenverkehr auf den Ports zugelassen wird 11001 (JMX-Exporteur) und 11002 (Node Exporter) innerhalb Ihrer Amazon-VPC, da der Scraper Zugriff auf diese DNS-Einträge benötigt, um Prometheus-Metriken zu sammeln.
- Die Amazon VPC, in der sich der Amazon MSK-Cluster befindet, muss DNS aktiviert.

> **Note**
>
> Der Cluster wird dem Scraper anhand seines Amazon-Ressourcennamens (ARN) zugeordnet. Wenn Sie einen Cluster löschen und dann einen neuen mit demselben Namen erstellen, wird der ARN für den neuen Cluster wiederverwendet. Aus diesem Grund versucht der Scraper, Metriken für den neuen Cluster zu sammeln. Du löscht Scraper getrennt vom Löschen des Clusters.

To create a scraper using the AWS API

Verwenden Sie den `CreateScraper` API-Operation, um einen Scraper mit dem zu erstellen AWS API. Im folgenden Beispiel wird ein Scraper in der Region USA Ost (Nord-Virginia) erstellt. Ersetzen Sie den *`example`* Geben Sie Ihre Amazon MSK-Cluster-Informationen ein und geben Sie Ihre Scraper-Konfiguration an.

> **Note**
>
> Konfigurieren Sie die Sicherheitsgruppe und die Subnetze so, dass sie Ihrem Zielcluster entsprechen. Schließen Sie mindestens zwei Subnetze in zwei Verfügbarkeitszonen ein.

```
               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
    }
}
```

In dem Beispiel ist der `scrapeConfiguration` Der Parameter erfordert eine Base64-kodierte Prometheus-Konfigurations-YAML-Datei, die die DNS-Einträge des MSK-Clusters angibt.

Jeder DNS-Eintrag stellt einen Broker-Endpunkt in einer bestimmten Availability Zone dar, sodass Kunden eine Verbindung zu Brokern herstellen können, die über die von Ihnen ausgewählten AZs verteilt sind, um eine hohe Verfügbarkeit zu gewährleisten.

Die Anzahl der DNS-Einträge in den Eigenschaften Ihres MSK-Clusters entspricht der Anzahl der Brokerknoten und Availability Zones in Ihrer Clusterkonfiguration:

- Standardkonfiguration — 3 Broker-Knoten auf 3 AZs = 3 DNS-Einträge
- Benutzerdefinierte Konfiguration — 2 Broker-Knoten auf 2 AZs = 2 DNS-Einträge

Um die DNS-Einträge für Ihren MSK-Cluster abzurufen, öffnen Sie die MSK-Konsole unter https://console.aws.amazon.com/msk/home? region=us-east-1#/start/. Gehen Sie zu Ihrem MSK-Cluster. Wählen Sie Eigenschaften, Makler, und Endpunkte.

Sie haben zwei Möglichkeiten, Prometheus so zu konfigurieren, dass Metriken aus Ihrem MSK-Cluster entfernt werden:

1. Cluster-level DNS-Auflösung (empfohlen) — Verwenden Sie den DNS-Basisnamen des Clusters, um automatisch alle Broker zu ermitteln. Wenn Ihr Broker-Endpunkt ist `b-1.clusterName.xxx.xxx.xxx`, verwenden `clusterName.xxx.xxx.xxx` als DNS-Eintrag. Dadurch kann Prometheus automatisch alle Broker im Cluster scrapen.

   Einzelne Broker-Endpunkte — Geben Sie jeden Broker-Endpunkt einzeln an, um eine detaillierte Steuerung zu ermöglichen. Verwenden Sie die vollständigen Broker-Identifikatoren (b-1, b-2) in Ihrer Konfiguration. Beispiel:

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

> **Note**
>
> Ersetzen `clusterName.xxx.xxx.xxx` mit Ihrem tatsächlichen MSK-Cluster-Endpunkt aus dem AWS Konsole.

Weitere Informationen finden Sie unter <dns_sd_config> in der Prometheus Dokumentation.

Das Folgende ist ein Beispiel für die Scraper-Konfigurationsdatei:

```
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}'
```

Führen Sie einen der folgenden Befehle aus, um die YAML-Datei in Base64 zu konvertieren. Sie können auch einen beliebigen Online-Base64-Konverter verwenden, um die Datei zu konvertieren.

Example Linux/macOS

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

Example Windows PowerShell

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

To create a scraper using the AWS CLI

Benutze das `create-scraper` Befehl zum Erstellen eines Schabers mit dem AWS Command Line Interface. Im folgenden Beispiel wird ein Scraper in der Region USA Ost (Nord-Virginia) erstellt. Ersetzen Sie den *example* Geben Sie Ihre Amazon MSK-Cluster-Informationen ein und geben Sie Ihre Scraper-Konfiguration an.

> **Note**
>
> Konfigurieren Sie die Sicherheitsgruppe und die Subnetze so, dass sie Ihrem Zielcluster entsprechen. Schließen Sie mindestens zwei Subnetze in zwei Verfügbarkeitszonen ein.

```
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'}"
```

- Im Folgenden finden Sie eine vollständige Liste der Scraper-Operationen, die Sie mit dem verwenden können AWS API:

  Erstellen Sie einen Scraper mit der CreateScraper API-Operation.

- Listen Sie Ihre vorhandenen Scraper mit der ListScrapers API-Operation auf.
- Aktualisieren Sie den Alias, die Konfiguration oder das Ziel eines Scrapers mit dem UpdateScraper API-Betrieb.
- Löschen Sie einen Scraper mit der DeleteScraper API-Operation.
- Erfahren Sie mehr über einen Scraper mit der DescribeScraper API-Operation.

Cross-account Einrichten

Gehen Sie wie folgt vor, um einen Scraper in einer kontoübergreifenden Konfiguration zu erstellen, wenn sich Ihr Amazon MSK-Cluster, von dem Sie Metriken sammeln möchten, in einem anderen Konto als der Amazon Managed Service for Prometheus Collector befindet.

Wenn Sie beispielsweise zwei Konten haben, das erste Quellkonto `account_id_source` wo sich das Amazon MSK befindet, und ein zweites Zielkonto `account_id_target` wo sich der Amazon Managed Service für Prometheus Workspace befindet.

Um einen Scraper in einer kontoübergreifenden Konfiguration zu erstellen

1. Erstelle im Quellkonto eine Rolle `arn:aws:iam::111122223333:role/Source` und fügen Sie die folgende Vertrauensrichtlinie hinzu.

```
{
    "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. Für jede Kombination aus Quelle (Amazon MSK-Cluster) und Ziel (Amazon Managed Service for Prometheus Workspace) müssen Sie eine Rolle erstellen `arn:aws:iam::444455556666:role/Target` und fügen Sie die folgende Vertrauensrichtlinie mit Berechtigungen für hinzu 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. Erstellen Sie einen Schaber mit dem `--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. Bestätigen Sie die Erstellung des Scrapers.

```
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>"
            }
        }
    ]
}
```

Wechselt zwischen RoleConfiguration und dienstbezogene Rolle

Wenn Sie zurück zu einer dienstverknüpften Rolle wechseln möchten, anstatt zu `RoleConfiguration` um in einen Amazon Managed Service for Prometheus-Workspace zu schreiben, müssen Sie den `UpdateScraper` und stelle einen Workspace im gleichen Konto wie der Scraper bereit, ohne `RoleConfiguration`. Das `RoleConfiguration` wird aus dem Scraper entfernt und die dienstgebundene Rolle wird verwendet.

Wenn Sie die Arbeitsbereiche im selben Konto wie der Scraper wechseln und den weiterhin verwenden möchten `RoleConfiguration`, müssen Sie erneut Folgendes angeben `RoleConfiguration` auf `UpdateScraper`.

Suchen und Löschen von Scrapern

Sie können das verwenden AWS API oder die AWS CLI um die Scraper in deinem Konto aufzulisten oder zu löschen.

> **Note**
>
> Stellen Sie sicher, dass Sie die neueste Version von verwenden AWS CLI oder SDK. Die neueste Version bietet Ihnen die neuesten Features und Funktionen sowie Sicherheitsupdates. Verwenden Sie alternativ AWS CloudShell, das automatisch ein stets aktuelles Befehlszeilenerlebnis bietet.

Die ListScrapers-API-Operation listet alle Scraper in Ihrem Konto auf.

Alternativ können Sie mit dem AWS CLI, rufen Sie an:

```
aws amp list-scrapers
```

`ListScrapers` gibt alle Scraper in Ihrem Konto zurück, zum Beispiel:

```
{
    "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"
                }
            }
        }
    ]
}
```

Suchen Sie zum Löschen eines Scrapers mithilfe der `ListScrapers`-Operation die `scraperId` für den Scraper, den Sie löschen möchten, und verwenden Sie dann die DeleteScraper-Operation, um ihn zu löschen.

Alternativ mit dem AWS CLI, rufen Sie an:

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

Von Amazon MSK gesammelte Metriken

Bei der Integration mit Amazon MSK entfernt der Amazon Managed Service for Prometheus Collector automatisch die folgenden Metriken:

Metriken: jmx_exporter- und pod_exporter-Jobs

Metrik	Beschreibung//Zweck
jmx_config_reload_failure_total	Gesamtzahl der Male, mit denen der JMX-Exporter seine Konfigurationsdatei nicht neu laden konnte.
jmx_scrape_duration_seconds	Zeit, die zum Scrapen von JMX-Metriken in Sekunden für den aktuellen Erfassungszyklus benötigt wurde.
jmx_scrape_error	Gibt an, ob beim JMX-Metrik-Scraping ein Fehler aufgetreten ist (1 = Fehler, 0 = Erfolg).

Metrik	Beschreibung//Zweck
java_lang_Memory_HeapMemoryUsage_used	Menge des Heap-Speichers (in Byte), der derzeit von der JVM verwendet wird.
java_lang_Memory_HeapMemoryUsage_max	Maximale Menge an Heap-Speicher (in Byte), die für die Speicherverwaltung verwendet werden kann.
java_lang_Memory_NonHeapMemoryUsage_used	Menge des Nicht-Heap-Speichers (in Byte), der derzeit von der JVM verwendet wird.
kafka_cluster_Partition_Value	Aktueller Status oder Wert in Bezug auf Kafka-Cluster-Partitionen, aufgeschlüsselt nach Partitions-ID und Thema.
kafka_consumer_consumer_coordinator_metrics_assigned_partitions	Anzahl der Partitionen, die diesem Consumer derzeit zugewiesen sind.
kafka_consumer_consumer_coordinator_metrics_commit_latency_avg	Durchschnittliche Zeit in Millisekunden, die für die Berechnung von Offsets benötigt wird.
kafka_consumer_consumer_coordinator_metrics_commit_rate	Anzahl der Offset-Commits pro Sekunde.
kafka_consumer_consumer_coordinator_metrics_failed_rebalance_total	Gesamtzahl der fehlgeschlagenen Neugewichte bei Verbrauchergruppen.
kafka_consumer_consumer_coordinator_metrics_last_heartbeat_seconds_ago	Anzahl der Sekunden, seit der letzte Herzschlag an den Koordinator gesendet wurde.
kafka_consumer_consumer_coordinator_metrics_rebalance_latency_avg	Durchschnittliche Zeit, die für die Neugewichtung von Verbrauchergruppen benötigt wird, in Millisekunden.
kafka_consumer_consumer_coordinator_metrics_rebalance_total	Gesamtzahl der Neugewichte der Verbrauchergruppen.

Metrik	Beschreibung//Zweck
kafka_consumer_consumer_fetch_manager_metrics_bytes_consumed_rate	Durchschnittliche Anzahl der vom Verbraucher pro Sekunde verbrauchten Byte.
kafka_consumer_consumer_fetch_manager_metrics_fetch_latency_avg	Durchschnittliche Zeit, die für eine Abrufanforderung in Millisekunden benötigt wird.
kafka_consumer_consumer_fetch_manager_metrics_fetch_rate	Anzahl der Abrufanforderungen pro Sekunde.
kafka_consumer_consumer_fetch_manager_metrics_records_consumed_rate	Durchschnittliche Anzahl der pro Sekunde verbrauchten Datensätze.
kafka_consumer_consumer_fetch_manager_metrics_records_lag_max	Maximale Verzögerung in Bezug auf die Anzahl der Datensätze für jede Partition in diesem Consumer.
kafka_consumer_consumer_metrics_connection_count	Aktuelle Anzahl aktiver Verbindungen.
kafka_consumer_consumer_metrics_incoming_byte_rate	Durchschnittliche Anzahl an Bytes, die pro Sekunde von allen Servern empfangen wurden.
kafka_consumer_consumer_metrics_last_poll_seconds_ago	Anzahl der Sekunden seit dem letzten Consumer-Aufruf von poll ().
kafka_consumer_consumer_metrics_request_rate	Anzahl der pro Sekunde gesendeten Anfragen.
kafka_consumer_consumer_metrics_response_rate	Anzahl der pro Sekunde empfangenen Antworten.
kafka_consumer_group_ConsumerLagMetrics_Wert	Aktueller Verbraucherverzögerungswert für eine Verbrauchergruppe, der angibt, wie weit der Verbraucher hinterherhinkt.
kafka_controller_KafkaController_Wert	Aktueller Status oder Wert des Kafka-Controllers (1 = aktiver Controller, 0 = nicht aktiv).

Metrik	Beschreibung//Zweck
kafka_controller_ControllerEventManager_Zählen	Gesamtzahl der verarbeiteten Controller-Ereignisse.
kafka_controller_ControllerEventManager_Gemein	Mittlere (durchschnittliche) Zeit, die für die Verarbeitung von Controller-Ereignissen benötigt wird.
kafka_controller_ControllerStats_MeanRate	Mittlere Rate der Controller-Statistikoperationen pro Sekunde.
kafka_koordinator_gruppe_GroupMetadataManager_Wert	Aktueller Status oder Wert des Gruppen-Metadaten-Managers für Verbrauchergruppen.
kafka_log_LogFlushStats_Zählen	Gesamtzahl der Log-Flush-Operationen.
kafka_log_LogFlushStats_Gemein	Mittlere (durchschnittliche) Zeit, die für Log-Flush-Operationen benötigt wurde.
kafka_log_LogFlushStats_MeanRate	Mittlere Rate von Log-Flush-Vorgängen pro Sekunde.
kafka_network_RequestMetrics_Zählen	Gesamtzahl der verarbeiteten Netzwerkanfragen.
kafka_network_RequestMetrics_Gemein	Mittlere (durchschnittliche) Zeit, die für die Bearbeitung von Netzwerkanfragen benötigt wurde.
kafka_network_RequestMetrics_MeanRate	Mittlere Rate von Netzwerkanfragen pro Sekunde.
kafka_network_Acceptor_MeanRate	Durchschnittliche Rate akzeptierter Verbindungen pro Sekunde.
kafka_server_Fetch_queue_size	Aktuelle Größe der Warteschlange für Abrufanfragen.

Metrik	Beschreibung//Zweck
kafka_server_Produce_queue_size	Aktuelle Größe der Warteschlange für Produktionsanfragen.
kafka_server_Request_queue_size	Aktuelle Größe der allgemeinen Anforderu ngswarteschlange.
kafka_server_BrokerTopicMetrics_Zählen	Gesamtzahl der Broker-Themenoperationen (Nachrichten) in/out, Byte in/out).
kafka_server_BrokerTopicMetrics_MeanRate	Durchschnittliche Rate der Transaktionen pro Sekunde bei Broker-Themen.
kafka_server_BrokerTopicMetrics_OneM inuteRate	One-minute gleitender Durchschnittskurs der Geschäfte im Zusammenhang mit Brokern.
kafka_server_DelayedOperationPurgatory_Wert	Aktuelle Anzahl verspäteter Operationen im Fegefeuer (die darauf warten, abgeschlossen zu werden).
kafka_server_DelayedFetchMetrics_MeanRate	Mittlere Rate verzögerter Abrufvorgänge pro Sekunde.
kafka_server_FetcherLagMetrics_Wert	Aktueller Lag-Wert für Replica-Fetcher-Threads (wie weit hinter dem Leader).
kafka_server_FetcherStats_MeanRate	Mittlere Rate der Abruf-Operationen pro Sekunde.
kafka_server_ReplicaManager_Wert	Aktueller Status oder Wert des Replikatm anagers.
kafka_server_ReplicaManager_MeanRate	Durchschnittliche Rate der Replica-Manager-Operationen pro Sekunde.
kafka_server_LeaderReplication_Byte_Rate	Rate der pro Sekunde replizierten Byte für Partitionen, bei denen dieser Broker führend ist.

Metrik	Beschreibung//Zweck
kafka_server_group_coordinator_metrics_group_completed_rebalance_count	Gesamtzahl der abgeschlossenen Neugewichte bei Verbrauchergruppen.
kafka_server_group_coordinator_metrics_offset_commit_count	Gesamtzahl der Offset-Commit-Operationen.
kafka_server_group_coordinator_metrics_offset_commit_rate	Rate der Offset-Commit-Operationen pro Sekunde.
kafka_server_socket_server_metrics_connection_count	Aktuelle Anzahl der aktiven Verbindungen.
kafka_server_socket_server_metrics_connection_creation_rate	Geschwindigkeit, mit der neue Verbindungen pro Sekunde hergestellt werden.
kafka_server_socket_server_metrics_connection_close_rate	Rate der Verbindungsschließungen pro Sekunde.
kafka_server_socket_server_metrics_failed_authentication_total	Gesamtzahl der fehlgeschlagenen Authentifizierungsversuche.
kafka_server_socket_server_metrics_incoming_byte_rate	Rate der eingehenden Byte pro Sekunde.
kafka_server_socket_server_metrics_outgoing_byte_rate	Rate der ausgehenden Byte pro Sekunde.
kafka_server_socket_server_metrics_request_rate	Rate der Anfragen pro Sekunde.
kafka_server_socket_server_metrics_response_rate	Rate der Antworten pro Sekunde.
kafka_server_socket_server_metrics_network_io_rate	Rate des Netzwerks I/O Operationen pro Sekunde.
kafka_server_socket_server_metrics_io_ratio	Bruchteil der Zeit, die in verbracht wurde I/O Operationen.

Metrik	Beschreibung//Zweck
kafka_server_controller_channel_metrics_connection_count	Aktuelle Anzahl aktiver Verbindungen für Controller-Kanäle.
kafka_server_controller_channel_metrics_incoming_byte_rate	Rate der eingehenden Byte pro Sekunde für Controller-Kanäle.
kafka_server_controller_channel_metrics_outgoing_byte_rate	Rate der ausgehenden Byte pro Sekunde für Controller-Kanäle.
kafka_server_controller_channel_metrics_request_rate	Rate der Anfragen pro Sekunde für Controller-Kanäle.
kafka_server_replica_fetcher_metrics_connection_count	Aktuelle Anzahl aktiver Verbindungen für Replica Fetcher.
kafka_server_replica_fetcher_metrics_incoming_byte_rate	Rate der eingehenden Byte pro Sekunde für den Replikatabruf.
kafka_server_replica_fetcher_metrics_request_rate	Rate der Anfragen pro Sekunde für den Replikatabruf.
kafka_server_replica_fetcher_metrics_failed_authentication_total	Gesamtzahl der fehlgeschlagenen Authentifizierungsversuche für den Replikatabruf.
kafka_server_ZooKeeperClientMetrics_Zählen	Gesamtzahl von ZooKeeper Client-Operationen.
kafka_server_ZooKeeperClientMetrics_Gemein	Mittlere Latenz von ZooKeeper Client-Operationen.
kafka_server_KafkaServer_Wert	Aktueller Status oder Wert des Kafka-Servers (zeigt in der Regel an, dass der Server läuft).
node_cpu_seconds_total	Gesamtzahl der Sekunden, die die CPUs in jedem Modus (Benutzer, System, Leerlauf usw.) verbracht haben, aufgeschlüsselt nach CPU und Modus.

Metrik	Beschreibung//Zweck
node_disk_read_bytes_total	Gesamtzahl der erfolgreich von Festplatten gelesenen Byte, aufgeschlüsselt nach Geräten.
node_disk_reads_completed_total	Gesamtzahl der erfolgreich abgeschlossenen Lesevorgänge für Festplatten, aufgeschlüsselt nach Geräten.
node_disk_writes_completed_total	Gesamtzahl der erfolgreich abgeschlossenen Schreibvorgänge für Festplatten, aufgeschl üsselt nach Geräten.
node_disk_written_bytes_total	Gesamtzahl der erfolgreich auf Festplatten geschriebenen Byte, aufgeschlüsselt nach Geräten.
node_filesystem_avail_bytes	Verfügbarer Speicherplatz im Dateisystem in Byte für Nicht-Root-Benutzer, aufgeschlüsselt nach Gerät und Einhängepunkt.
node_filesystem_size_bytes	Gesamtgröße des Dateisystems in Byte, aufgeschlüsselt nach Gerät und Einhängep unkt.
node_filesystem_free_bytes	Freier Speicherplatz im Dateisystem in Byte, aufgeschlüsselt nach Gerät und Einhängep unkt.
node_filesystem_files	Gesamtzahl der Dateiknoten (Inodes) im Dateisystem, aufgeschlüsselt nach Gerät und Einhängepunkt.
node_filesystem_files_free	Anzahl der freien Dateiknoten (Inodes) im Dateisystem, aufgeschlüsselt nach Gerät und Einhängepunkt.

Metrik	Beschreibung//Zweck
node_filesystem_readonly	Gibt an, ob das Dateisystem schreibgeschützt eingehängt ist (1 = schreibgeschützt, 0 = lesen/schreiben).
node_filesystem_device_error	Gibt an, ob beim Abrufen der Dateisyst emstatistiken ein Fehler aufgetreten ist (1 = Fehler, 0 = Erfolg).

Erfassen Sie Kennzahlen zu Amazon MSK-Verbrauchergruppen

Mit einem verwalteten Kafka-Exporteur erfasst ein Amazon Managed Service for Prometheus Collector Kennzahlen zu Verbrauchergruppen aus Ihrem Amazon MSK-Cluster, wie z. B. Consumer Lag, Committed Offsets und die Anzahl der Mitglieder in jeder Verbrauchergruppe. Amazon Managed Service für Prometheus führt den Exporter für Sie aus, sodass Sie nicht selbst einen Exporter installieren oder betreiben müssen. Ihr Ziel kann ein Amazon Managed Service für Prometheus Workspace oder Amazon sein CloudWatch. Diese Metriken ergänzen die JMX- und Node Exporter-Metriken, die der Collector aus Open Monitoring abruft.

Der Scraper stellt eine Verbindung zu Ihrem Cluster her mit AWS Authentifizierung für Identitäts- und Zugriffsmanagement (IAM). Sie gewähren dem Scraper mithilfe einer Amazon MSK-Cluster-Richtlinie Zugriff auf den Cluster. Um Kennzahlen für Verbrauchergruppen zu sammeln, erstellen Sie eine Scrape-Konfiguration, erstellen Sie einen Scraper und gewähren Sie dem Scraper dann Zugriff auf Ihren Cluster, wie in den folgenden Abschnitten beschrieben.

Informationen zu den Metriken finden Sie unter Consumer-group Metriken. Um die Metriken an Amazon zu senden CloudWatch, siehe Erfassen Sie Kennzahlen zu Amazon MSK-Verbrauchergruppen in CloudWatch.

Voraussetzungen

Für die Erfassung von Kennzahlen zu Verbrauchergruppen gibt es einige Voraussetzungen:

- Ein aktiver, von Amazon MSK bereitgestellter Cluster mit Standard-Brokern und Apache Kafka Version 3.6.0 oder höher, im selben Konto und AWS Region als Scraper.

- Die IAM-Zugriffskontrolle wurde für den Cluster aktiviert. Sie können die TLS-Client-Authentifizierung gleichzeitig aktiviert lassen, sodass Clients, die Zertifikate verwenden, weiterhin eine Verbindung herstellen können.
- Mindestens zwei Subnetze in verschiedenen Availability Zones in der Amazon VPC des Clusters, wobei DNS aktiviert ist.
- Sicherheitsgruppen, die es dem Scraper ermöglichen, die privaten IAM-Broker-Endpunkte des Clusters auf Port 9098 zu erreichen. Fügen Sie der Sicherheitsgruppe des Clusters eine Regel für eingehenden Datenverkehr hinzu, die TCP-Verkehr auf Port 9098 aus der Sicherheitsgruppe zulässt, die Sie für den Scraper angeben.
- Ein Amazon Managed Service für den Prometheus-Workspace in der Region des Scrapers und ein AWS CLI Version, die unterstützt `kafkaConfiguration` in `create-scraper` und `update-scraper`.
- Erlaubnis, einen Scraper zu erstellen oder zu aktualisieren und den Cluster zu validieren: `kafka:DescribeClusterV2`, `kafka:GetBootstrapBrokers`, `kafka-cluster:Connect`, und `kafka-cluster:DescribeTopic`. Um die Cluster-Richtlinie hinzuzufügen, benötigen Sie außerdem `kafka:GetClusterPolicy` und `kafka:PutClusterPolicy`. Die Berechtigungen zum Erstellen der serviceverknüpften Rolle des Scrapers finden Sie unter Service-linked Rollenberechtigungen.

Der Scraper verwendet seine dienstgebundene Rolle, um eine Verbindung zum Cluster herzustellen, sodass Sie dem Exporteur keine Rolle zuweisen. Standardmäßig hat die Rolle keine Berechtigungen für Ihren Cluster. Sie gewähren ihnen Gewähren Sie dem Scraper Zugriff auf den Cluster.

Das verwaltete Kafka-Exportprogramm unterstützt die folgenden Cluster nicht:

- Serverlose Amazon MSK-Cluster. Serverlose Cluster akzeptieren keine Cluster-Richtlinien mit `kafka-cluster` Aktionen. Benutze die Metriken, an die Amazon MSK sendet CloudWatch stattdessen.
- Cluster, bei denen die IAM-Authentifizierung nicht aktiviert ist.
- Cluster in einem anderen Konto oder einer anderen Region als der Scraper.

Die Open Monitoring- und Broker-Ports 11001 und 11002 werden nur benötigt, wenn Sie auch JMX- und Node Exporter-Metriken erfassen.

Erstellen Sie eine Scrape-Konfiguration

Sie geben eine Scrape-Konfiguration an, wenn Sie den Scraper erstellen. Der verwaltete Collector stellt eine Verbindung zu dem von Ihnen angegebenen Cluster her und erfasst seine Metriken automatisch, sodass Sie in der Konfiguration keine Scrape-Ziele angeben. Die Konfiguration muss eine enthalten `scrape_configs` Abschnitt mit einem Job, dessen `job_name` ist genau `kafka-exporter`. Eine Konfiguration, die keinen Job mit dem Namen enthält `kafka-exporter` wird abgelehnt. Der Kollektor legt das Scrape-Intervall für diesen Job fest.

Verwenden Sie die Scrape-Konfiguration, um Labels hinzuzufügen und die gesammelten Metriken zu filtern. Im folgenden Beispiel werden „Consumer Lag“, „Committed Offsets“ und die Gruppenzugehörigkeit beibehalten. Es fügt auch eine hinzu `clusterArn` Bezeichnung, damit Sie Cluster bei der Abfrage unterscheiden können. Ersetzen Sie den Cluster-ARN durch den ARN Ihres Clusters und speichern Sie die Konfiguration unter `msk-consumer.yaml`. Verwenden Sie denselben Cluster-ARN, wenn Sie den Scraper erstellen.

```
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
```

Der Exporter kann auch andere Metriken senden, z. B. Themen- und Partitions-Offsets sowie Partitionszahlen. Amazon MSK stellt einige dieser Informationen bereits über Open Monitoring und die Metriken, an die Amazon MSK sendet CloudWatch. Um weitere Export-Metriken zu sammeln, fügen Sie sie dem `regex`, zum Beispiel `kafka_consumergroup_(current_offset|members|lag)|kafka_topic_partitions`. Jede Metrik, die Sie hinzufügen, erhöht die Anzahl der Serien, die Sie aufnehmen.

Um den Exporter zu einem vorhandenen Scraper hinzuzufügen, fügen Sie den `kafka-exporter` Stellen Sie den Job auf die vollständige Scrape-Konfiguration des Scrapers um und behalten Sie die anderen Jobs bei. Aktualisieren Sie dann den Scraper wie unter beschrieben Aktualisieren oder entfernen Sie den Exporter. Sie können Metriken in diesem Job umbenennen, aber Sie können keine

Umbenennungsregeln verwenden, um zu ändern, wie der Collector eine Verbindung zum Exporter herstellt.

Einen Scraper erstellen

Um dem Scraper mitzuteilen, von welchem Cluster die Daten erfasst werden sollen, geben Sie den Cluster im `exporters` Feld der Anfrage. Geben Sie für einen Amazon MSK-Cluster einen `kafkaConfiguration` mit dem `clusterArn` des Clusters. Sie können auch eine festlegen `topicFilter` und ein `consumerGroupFilter` um Themen und Verbrauchergruppen namentlich auszuwählen. Sie stellen das Netzwerk (Subnetze und Sicherheitsgruppe) separat in der `source` Feld.

Benutze das `create-scraper` Befehl zum Erstellen eines Schabers mit dem AWS Command Line Interface. Im folgenden Beispiel werden Themen ausgewählt, die mit beginnen `orders` und Verbrauchergruppen, die beginnen mit `payments`. Ersetzen Sie die *`example`* Inhalt, einschließlich beider Filter, durch Ihre eigenen Cluster-, Netzwerk-, Workspace- und Filterwerte. Um alle Themen oder alle Verbrauchergruppen auszuwählen, lassen Sie den Filter weg.

> **Note**
>
> Konfigurieren Sie die Sicherheitsgruppe und die Subnetze so, dass sie der Amazon VPC Ihres Amazon MSK-Clusters entsprechen. Schließen Sie mindestens zwei Subnetze in zwei Availability Zones ein.

```
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"}}'
```

Beide Filter sind optional. Wenn Sie einen Filter weglassen, wählt der Scraper alle Themen oder alle Verbrauchergruppen aus. Jeder Filter muss ein regulärer Ausdruck sein, der beginnt mit ^ und endet mit $, mit einer maximalen Länge von 256 Zeichen. Ein Scraper kann einen Exporteur haben.

Der `topicFilter` wählt die Themen aus, für die das Exportprogramm Themenmetriken liest. Consumer-group Metriken können auch andere Themen beinhalten, die von einer ausgewählten Verbrauchergruppe konsumiert werden. Um zu kontrollieren, welche Kennzahlen Ihr Ziel erreichen, verwenden Sie `metric_relabel_configs` in deiner Scrape-Konfiguration.

Notieren Sie sich die Scraper-ID. Überprüfen Sie den Status mit dem folgenden Befehl und fahren Sie fort, wenn der Status lautet `ACTIVE`:

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

Der Scraper sammelt keine Kafka-Metriken, bis Sie ihm Zugriff auf den Cluster gewähren.

Gewähren Sie dem Scraper Zugriff auf den Cluster

Der Scraper benötigt die Erlaubnis, eine Verbindung zu Ihrem Cluster herzustellen und dessen Themen- und Verbrauchergruppen-Metadaten zu lesen. Fügen Sie eine Amazon MSK-Cluster-Richtlinie hinzu, die der serviceverknüpften Rolle des Scrapers die folgenden Aktionen ermöglicht:

- `kafka-cluster:Connect` auf dem Cluster
- `kafka-cluster:DescribeTopic` zu seinen Themen
- `kafka-cluster:DescribeGroup` zu ihren Verbrauchergruppen

Ohne diese Politik ist der Scraper `ACTIVE` sammelt aber keine Kafka-Metriken. Nachdem Sie die Richtlinie hinzugefügt haben, beginnt der Scraper automatisch mit der Erfassung, und es kann einige Minuten dauern, bis die ersten Metriken angezeigt werden. Sie müssen den Scraper nicht aktualisieren.

Gewähren Sie dem Scraper die servicebasierte Rolle, auch wenn der Scraper Folgendes verwendet `roleConfiguration` um Metriken an einen Workspace in einem anderen Konto zu senden. Gewähren Sie weder der Quellrolle noch der Zielrolle.

Gewähren Sie einen Scraper

Ermitteln Sie den ARN der Rolle des Scrapers mit dem Folgenden AWS CLI Befehl:

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

Jeder Scraper hat seine eigene Rolle, `AWSServiceRoleForAmazonPrometheusScraper_`*`suffix`*. Wenn Sie einen Scraper neu erstellen, gewähren Sie ihm die neue Rolle.

Speichern Sie die folgende Richtlinie unter `policy.json`. Ersetzen Sie den Rollen-ARN, die Region, die Konto-ID, den Clusternamen und die Cluster-ID.

```
{
  "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/*"
    }
  ]
}
```

Um einen weiteren Scraper zu gewähren, fügen Sie die drei Anweisungen erneut mit dem Rollen-ARN dieses Scrapers hinzu. Um weniger Themen oder Verbrauchergruppen zuzulassen,

bearbeiten Sie die Richtlinie wie unter beschrieben Beschränken Sie die Anzahl der Themen und Verbrauchergruppen bevor Sie es anwenden.

Wenden Sie die Richtlinie an

> **Important**
>
> Verwenden Sie keinen Root-Principal für das Konto, wie z. B. `arn:aws:iam::111122223333:root`. Amazon MSK akzeptiert es, gewährt dem Scraper aber nichts. Verwenden Sie keinen Platzhalter in einer Rollen-ARN als Prinzipal. Amazon MSK lehnt es mit dem Fehler ab `The cluster policy is not valid`. Um jeden Scraper auf dem Konto zu gewähren, siehe Optional: Gewähre jedem Scraper im Konto.

Um die Cluster-Richtlinie zu lesen und zu ändern, benötigen Sie `kafka:GetClusterPolicy` und `kafka:PutClusterPolicy`.

So fügen Sie die Cluster-Richtlinie hinzu

1. Prüfen Sie, ob der Cluster bereits über eine Richtlinie verfügt:

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

   EIN `NotFoundException` Ein Fehler bedeutet, dass der Cluster keine Richtlinie hat.

2. Wenn der Cluster keine Richtlinie hat, fügen Sie Ihre hinzu:

   ```
   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. Wenn der Cluster bereits über eine Richtlinie verfügt, führen Sie die Anweisungen zusammen. `put-cluster-policy` ersetzt das gesamte Richtliniendokument. Das `Policy` Wert von `get-cluster-policy` ist die bestehende Richtlinie als JSON-Zeichenfolge. Fügen Sie ihre Anweisungen dem hinzu `Statement` liste in `policy.json`, sodass die Datei sowohl die

vorhandenen als auch die neuen Anweisungen enthält. Dann übergebe den `CurrentVersion` Wert von `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
```

Optional: Gewähre jedem Scraper auf dem Konto

Wir empfehlen, dass Sie Zugriff auf die ARNs der einzelnen Scraper-Rollen gewähren. Wenn Sie jedem aktuellen und zukünftigen Scraper im Konto gewähren müssen, können Sie stattdessen die folgende Richtlinie verwenden. Du kannst ihn hinzufügen, bevor du einen Scraper erstellst, und er funktioniert weiter, wenn du einen ersetzst. Behalten Sie beide Bedingungen in jeder Anweisung bei. Speichern Sie die folgende Richtlinie unter `policy.json`. Ersetzen Sie die Region, die Konto-ID, den Clusternamen und die Cluster-ID in jeder Anweisung und wenden Sie sie wie unter beschrieben an Wenden Sie die Richtlinie an.

```
{
  "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_*" }
      }
    }
  ]
}
```

Diese Richtlinie erlaubt jeden aktuellen und zukünftigen Scraper auf dem Konto. Das `aws:PrincipalAccount` und `ArnLike` Bedingungen sind es, die es einschränken. Entferne keine der beiden Bedingungen.

Beschränken Sie die Themen und Verbrauchergruppen

Beide Richtlinien gewähren allen Themen und Verbrauchergruppen im Cluster Unterstützung. Um weniger zu gewähren, ersetzen Sie den letzten * zum Beispiel in den Themen- und Gruppen-ARNs `topic/my-cluster/cluster-id/orders-*`. Diese ARN-Muster sind IAM-Platzhalter, nicht `topicFilter` oder `consumerGroupFilter` reguläre Ausdrücke. Wenn die Richtlinie weniger Themen oder Gruppen zulässt, als die Filter auswählen, meldet der Scraper weniger Themen und Offsets.

Um eine Richtlinie einzugrenzen, die Sie bereits angewendet haben, ersetzen Sie Ihre früheren Aussagen durch die eingegrenzten. Führen Sie sie nicht zusammen, da die früheren Aussagen immer noch für alle Themen und Gruppen gelten.

Überprüfen Sie die Erfassung der Kennzahlen

Um zu bestätigen, dass der Scraper Metriken sammelt, führen Sie jede der folgenden PromQL-Abfragen separat für Ihren Workspace aus. Weitere Informationen finden Sie unter Fragen Sie Ihre Prometheus-Metriken ab. Ersetzen Sie den Cluster-ARN. Metriken werden nur für Verbrauchergruppen angezeigt, die Ihrem Verbrauchergruppenfilter entsprechen und die Mitglieder oder festgelegte Offsets haben.

```
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"}
```

Es kann einige Minuten dauern, bis die ersten Metriken angezeigt werden, nachdem Sie die Cluster-Richtlinie hinzugefügt haben. Wenn danach keine Metriken angezeigt werden, überprüfen Sie Folgendes. Ein leeres Ergebnis bedeutet nicht, dass die Verzögerung beim Verbraucher gleich Null ist.

- Wenn der Scraper ist `ACTIVE`, aber `up{job="kafka-exporter"}` ist `0` und nein `kafka_*` Metriken werden angezeigt. Überprüfen Sie die Cluster-Richtlinie. Eine erfolgreiche Erstellungs- oder Aktualisierungsanforderung bedeutet nicht, dass der Scraper auf den Cluster zugreifen kann. Führen Sie aus `aws kafka get-cluster-policy`, und bestätigen Sie, dass die Versicherung gewährt `kafka-cluster:Connect`, `kafka-cluster:DescribeTopic`, und `kafka-cluster:DescribeGroup` zur aktuellen dienstbezogenen Rolle ARN des Scrapers. Wenn Sie den Scraper neu erstellt haben, hat sich sein Rollen-ARN geändert. Wenn Sie die kontoweite Richtlinie verwenden, vergewissern Sie sich, dass in jeder Kontoauszug beide Angaben enthalten sind `StringEquals` auf `aws:PrincipalAccount` und `ArnLike` auf `aws:PrincipalArn`. Sie müssen den Scraper nicht aktualisieren, nachdem Sie die Richtlinie repariert haben.
- Wenn die Erstellung oder Aktualisierung mit einem Berechtigungsfehler fehlschlägt, geben Sie dem Aufrufer die Scraper- und Cluster-Berechtigungen in Voraussetzungen, dann versuchen Sie den Befehl erneut.
- Wenn der Scraper keine Verbindung zu den Brokern herstellen kann, aktivieren Sie die IAM-Authentifizierung für den Cluster. Überprüfen Sie das VPC-DNS und ermöglichen Sie der Sicherheitsgruppe des Scrapers, die privaten IAM-Broker-Endpunkte auf Port 9098 zu erreichen.
- Wenn der Scraper ist `ACTIVE` aber die Abfragen geben keine Metriken zurück, überprüfe das `consumerGroupFilter` entspricht einer vorhandenen Verbrauchergruppe. Behalten Sie die Namen der Metriken und `clusterArn`, `consumergroup`, `topic`, und `partition` Beschriftungen in der Scrape-Konfiguration. Verwenden Sie denselben Cluster-ARN in der Konfiguration und Abfrage. Wenn die Clusterrichtlinie weniger Themen oder Gruppen zulässt, als die Filter auswählen, werden weniger Themen und Offsets angezeigt.

- Wenn die Cluster-Richtlinie korrekt ist, aber `up{job="kafka-exporter"}` ist immer noch `0` Versuchen Sie es bei einem Cluster mit vielen Themen und Verbrauchergruppen einzugrenzen `topicFilter` und `consumerGroupFilter`.

Aktualisiere oder entferne den Exporter

Um die Filter zu ändern, aktualisieren Sie den Scraper mit seiner vollständigen Scrape-Konfiguration. Mit dem folgenden Befehl werden ein Thema und eine Verbrauchergruppe ausgewählt. Behalte das `kafka-exporter` Job und der `clusterArn` Etikett in Ihrer Scrape-Konfiguration.

```
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')"
```

Weglassen `exporters` bei einem Update bleibt der aktuelle Exporter erhalten. Um ihn zu entfernen, übergeben Sie eine leere Liste, `--exporters '[]'`, und senden Sie die vollständige Scrape-Konfiguration mit dem `kafka-exporter` Job entfernt. Behalte alle anderen Jobs, die du noch brauchst.

Folgendes AWS CLI Der Befehl löscht den Scraper und stoppt seine gesamte Metriksammlung. Der Amazon MSK-Cluster oder der Ziel-Workspace werden nicht gelöscht.

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

Entfernen Sie den Zugriff des Scrapers auf den Cluster

Durch das Entfernen der Cluster-Richtlinie wird die Erfassung nicht sofort beendet. Um die Erfassung zu beenden, entfernen Sie zuerst den Exporter oder löschen Sie den Scraper. Entfernen Sie dann die Grundsatzerklärungen.

Um den Zugriff des Scrapers zu entfernen

1. Notieren Sie sich die ARN der Rolle des Scrapers. Entfernen Sie dann den Exporter oder löschen Sie den Scraper, wie weiter oben in diesem Abschnitt beschrieben.
2. Führen Sie `aws amp describe-scraper`. Vergewissern Sie sich, dass der Scraper-Status lautet `ACTIVE` und dass der Scraper keinen Exporteur hat oder dass der Scraper nicht mehr existiert. Andere Scraper können weiterhin Metriken für denselben Cluster senden.
3. Führen Sie `aws kafka get-cluster-policy`. Entferne nur die Statements, die du für den Scraper hinzugefügt hast. Behalten Sie Kontoauszüge bei, die anderen Kunden Zugriff gewähren. Wenn Sie die kontoweite Richtlinie verwendet haben und andere Scraper weiterhin Zugriff benötigen, ersetzen Sie ihre Anweisungen durch Anweisungen, die den Rollen-ARN jedes Scrapers benennen, der den Zugriff behält.
4. Führen Sie aus `aws kafka put-cluster-policy` mit den restlichen Aussagen und dem `CurrentVersion` Wert, wie in Wenden Sie die Richtlinie an. Wenn keine Kontoauszüge mehr vorhanden sind, löschen Sie die Richtlinie stattdessen mit `aws kafka delete-cluster-policy`. Dieser Befehl benötigt die `kafka:DeleteClusterPolicy` Erlaubnis.

Consumer-group Metriken

Wenn Sie Kennzahlen für Verbrauchergruppen mit der Beispielkonfiguration unter sammeln Erfassen Sie Amazon MSK-Metriken für Verbrauchergruppen, sendet der Amazon Managed Service for Prometheus Collector die folgenden Metriken. Jede Metrik hat die `clusterArn` Bezeichnung aus der Scrape-Konfiguration, sodass Sie Cluster unterscheiden können.

Metrik	Beschreibung/Zweck
kafka_consumergroup_lag	Unterschied zwischen dem neuesten Offset in einer Partition und dem festgeschriebenen Offset der Consumer-Gruppe. Dies ist ungefähr die Anzahl der Nachrichten, für die die Gruppe noch keinen Commit ausgeführt hat. Beschriftungen: `consumergroup` , `topic`, `partition` .
kafka_consumergroup_current_offset	Letzter Offset, den die Verbrauchergruppe für eine Partition festgeschrieben hat.

Metrik	Beschreibung/Zweck
	Beschriftungen: `consumergroup` , `topic`, `partition` .
Mitglieder der kafka_consumergroup_	Anzahl der Mitglieder der Verbrauchergruppe. Etikett: `consumergroup` .

Ein Wert von `-1` for lag oder committed offset bedeutet, dass die Consumer-Gruppe noch keinen Offset für diese Partition festgeschrieben hat. Lassen Sie diese Werte weg, wenn Sie die Verzögerung addieren. Die folgende Abfrage zeigt beispielsweise die Gesamtverzögerung jeder Verbrauchergruppe für jedes Thema:

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

Diese Summe beinhaltet keine Partitionen, für die die Consumer-Gruppe keinen Offset festgeschrieben hat.

Das Exportprogramm berechnet die Verzögerung anhand der Offsets, die es aus Ihrem Cluster liest. Seine Werte können von den abweichen Kennzahlen zur Verzögerung der Verbraucher, die Amazon MSK bereitstellt, weil jeder Offsets zu unterschiedlichen Zeiten liest.

Einschränkungen

Die aktuelle Amazon MSK-Integration mit Amazon Managed Service für Prometheus hat die folgenden Einschränkungen:

- Wird nur für von Amazon MSK bereitgestellte Cluster unterstützt (nicht verfügbar für Amazon MSK Serverless)
- Nicht unterstützt für Amazon MSK-Cluster mit aktiviertem öffentlichen Zugriff in Kombination mit dem Kraft-Metadatenmodus
- Unterstützt derzeit eine 1:1 -Zuordnung zwischen Amazon MSK-Clustern und Amazon Managed Service for Prometheus collectors/workspaces

Richten Sie verwaltete Prometheus-Collectors für Amazon ein OpenSearch Bedienung

Der Amazon Managed Service für Prometheus verwaltete Collector für Amazon OpenSearch Der Service scrapt automatisch Prometheus-compatible Metriken von Ihrem OpenSearch Service-Domain und leitet sie an Ihr Ziel weiter. Amazon Managed Service for Prometheus verwaltet den Collector für Sie und bietet Ihnen die Skalierbarkeit, Sicherheit und Zuverlässigkeit, die Sie benötigen, ohne dass Sie Instanzen, Agenten oder Scraper selbst verwalten müssen. Ihr Ziel kann ein Amazon Managed Service für Prometheus Workspace oder Amazon sein CloudWatch.

Sie können auch einen Scraper erstellen, der in Amazon Elastic Kubernetes Service oder in Amazon Managed Streaming für Apache Kafka integriert ist. Weitere Informationen finden Sie unter Integrieren Sie Amazon EKS und Integrieren Sie Amazon MSK.

Einen Scraper erstellen

Bei diesem Verfahren wird davon ausgegangen, dass Sie mit Amazon vertraut sind OpenSearch Verwaltung der Servicedomäne und Netzwerkkonzepte von Amazon Virtual Private Cloud.

Es gibt einige Voraussetzungen für die Erstellung eines Scrapers für OpenSearch Dienstdomäne:

- Ein Amazon OpenSearch Servicedomäne mit Zugriff auf Amazon Virtual Private Cloud. Nur Unterstützung für verwaltete Sammler OpenSearch Servicedomänen, die VPC-Zugriff haben. Domänen mit öffentlichem Zugriff werden nicht unterstützt.
- Sicherheitsgruppen, die es dem verwalteten Collector ermöglichen, Ihre zu erreichen OpenSearch Service-Domain-Endpunkt über HTTPS (Port 443). Fügen Sie der Sicherheitsgruppe der Domain eine Regel für eingehenden Datenverkehr hinzu, die HTTPS-Verkehr aus der Sicherheitsgruppe zulässt, die Sie für den Collector bereitstellen.

Um dem Scraper zu sagen, welcher OpenSearch Dienstdomäne, von der die Daten abgeholt werden sollen, geben Sie die Domäne in der `exporters` Feld der Anfrage. Das `exporters` Das Feld enthält eine Liste von Exportkonfigurationen; für eine OpenSearch Dienstdomäne, geben Sie eine an `openSearchConfiguration` mit dem `domainArn` der Domain. Sie stellen das Netzwerk (Subnetze und Sicherheitsgruppe) separat in der `source` Feld, wie in den folgenden Beispielen gezeigt.

> **Note**
>
> Der Scraper verwendet den VPC-Endpunkt der Domain, der beim Erstellen des Scrapers aufgelöst wurde. Wenn Sie die Domain neu erstellen, erhält sie einen neuen Endpunkt. Sie müssen also einen neuen Scraper erstellen oder den vorhandenen aktualisieren mit UpdateScraper um daraus zu sammeln.

Sie geben eine Scrape-Konfiguration an, wenn Sie den Scraper erstellen. Der verwaltete Collector stellt eine Verbindung zu der von Ihnen angegebenen Domain her und erfasst seine Metriken automatisch, sodass Sie in der Konfiguration keine Scrape-Ziele angeben. Die Konfiguration muss eine enthalten `scrape_configs` Abschnitt mit einem Job, dessen `job_name` ist genau `opensearch-exporter`. Eine Konfiguration, die keinen Job mit dem Namen enthält `opensearch-exporter` wird abgelehnt. Verwenden Sie die Scrape-Konfiguration, um das Scrape-Intervall festzulegen und optional die gesammelten Metriken zu filtern oder neu zu kennzeichnen. Im Folgenden finden Sie ein Beispiel für eine Scrape-Konfiguration:

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

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

Weitere Informationen zu den unterstützten Scrape-Konfigurationsoptionen finden Sie unter Scraper-Konfiguration.

To create a scraper using the AWS API

Benutze das `CreateScraper` API-Operation, um einen Scraper mit dem zu erstellen AWS API. Im folgenden Beispiel wird ein Scraper in der Region USA Ost (Nord-Virginia) erstellt, der Metriken von einer OpenSearch Service-Domain und sendet sie an einen Amazon Managed Service für Prometheus Workspace. Ersetzen Sie das *`example`* Inhalte mit deinen eigenen Domain-, Netzwerk- und Workspace-Informationen und gib deine Scraper-Konfiguration an.

> **Note**
>
> Konfigurieren Sie die Sicherheitsgruppe und die Subnetze so, dass sie der Amazon-VPC Ihres OpenSearch Service-Domäne. Schließen Sie mindestens zwei Subnetze in zwei Availability Zones ein.

```
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"
    }
}
```

Um die Metriken an Amazon zu senden CloudWatch Ersetzen Sie anstelle eines Amazon Managed Service für Prometheus Workspace den `destination` mit einem CloudWatch Konfiguration:

```
    "destination": {
```

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

Die `scrapeConfiguration` Der Parameter erfordert eine Base64-kodierte Prometheus-Konfigurations-YAML-Datei. Führen Sie einen der folgenden Befehle aus, um die YAML-Datei in Base64 zu konvertieren. Sie können auch einen beliebigen Online-Base64-Konverter verwenden, um die Datei zu konvertieren.

Example Linux/macOS

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

Example Windows PowerShell

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

To create a scraper using the AWS CLI

Benutze das `create-scraper` Befehl zum Erstellen eines Schabers mit dem AWS Command Line Interface. Im folgenden Beispiel wird ein Scraper in der Region USA Ost (Nord-Virginia) erstellt. Ersetzen Sie den *example* Inhalte mit deinen eigenen Domain-, Netzwerk- und Workspace-Informationen und gib deine Scraper-Konfiguration an.

> **Note**
>
> Konfigurieren Sie die Sicherheitsgruppe und die Subnetze so, dass sie der Amazon-VPC Ihres OpenSearch Service-Domäne. Schließen Sie mindestens zwei Subnetze in zwei Availability Zones ein.

```
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"}}'
```

- Im Folgenden finden Sie eine vollständige Liste der Scraper-Operationen, die Sie mit dem verwenden können AWS API:

  Erstellen Sie einen Scraper mit der CreateScraper API-Operation.
- Listen Sie Ihre vorhandenen Scraper mit der ListScrapers API-Operation auf.
- Aktualisieren Sie den Alias, die Konfiguration oder das Ziel eines Scrapers mit dem UpdateScraper API-Betrieb.
- Löschen Sie einen Scraper mit der DeleteScraper API-Operation.
- Erfahren Sie mehr über einen Scraper mit der DescribeScraper API-Operation.

Cross-account Einrichten

Um einen Scraper in einem kontoübergreifenden Setup zu erstellen, wenn OpenSearch Die Service-Domain, für die Sie Metriken sammeln möchten, befindet sich in einem anderen Konto als der Amazon Managed Service for Prometheus Collector. Gehen Sie wie folgt vor.

Sie haben beispielsweise zwei Konten: ein Quellkonto `account_id_source` wo OpenSearch Die Dienstdomäne befindet sich und ein Zielkonto `account_id_target` wo sich der Amazon Managed Service für Prometheus Workspace befindet.

> **Note**
>
> Die Vertrauensrichtlinien in den folgenden Schritten beziehen sich auf den Scraper anhand seines ARN, der Folgendes beinhaltet *`scraper-id`* das existiert erst, wenn Sie den Scraper erstellen. Um dieses Ordnungsproblem zu vermeiden, erstellen Sie zuerst die Rollen mit einem Platzhalter (`scraper/*`) erstellen Sie anstelle des spezifischen Scraper-ARN den Scraper und aktualisieren Sie dann beide Vertrauensrichtlinien, um den Platzhalter durch den tatsächlichen Scraper-ARN zu ersetzen `CreateScraper` kehrt zurück.

Um einen Scraper in einem kontoübergreifenden Setup zu erstellen

1. Erstelle im Quellkonto eine Rolle `arn:aws:iam::`*`111122223333`*`:role/Source` und fügen Sie die folgende Vertrauensrichtlinie hinzu.

```
{
    "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. Erstellen Sie im Zielkonto eine Rolle `arn:aws:iam::444455556666:role/Target` und fügen Sie die folgende Vertrauensrichtlinie hinzu, die es der Quellrolle ermöglicht, sie zu übernehmen.

```
{
  "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"
      }
  }
}
```

Fügen Sie der Zielrolle eine Berechtigungsrichtlinie hinzu, die es ihr ermöglicht, an Ihr Ziel zu schreiben. Wenn es sich bei Ihrem Ziel um einen Amazon Managed Service für Prometheus Workspace handelt, fügen Sie das Dokument als Anhang an AmazonPrometheusRemoteWriteAccess (welche Zuschüsse `aps:RemoteWrite`).

Wenn Ihr Ziel Amazon ist CloudWatch, fügen Sie eine Police bei, die Folgendes gewährt `cloudwatch:PutMetricData` auf dem Datensatz.

3. Erstellen Sie einen Scraper mit dem `--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. Bestätigen Sie die Erstellung des Scrapers.

```
aws amp list-scrapers
```

Suchen und Löschen von Scrapern

Sie können das verwenden AWS API oder die AWS CLI um die Scraper in deinem Konto aufzulisten oder zu löschen.

> **Note**
>
> Stellen Sie sicher, dass Sie die neueste Version von verwenden AWS CLI oder SDK. Die neueste Version bietet Ihnen die neuesten Features und Funktionen sowie Sicherheitsupdates. Verwenden Sie alternativ AWS CloudShell, das automatisch ein stets aktuelles Befehlszeilenerlebnis bietet.

Die ListScrapers-API-Operation listet alle Scraper in Ihrem Konto auf. Alternativ können Sie mit dem AWS CLI, rufen Sie an:

```
aws amp list-scrapers
```

Suchen Sie zum Löschen eines Scrapers mithilfe der `ListScrapers`-Operation die `scraperId` für den Scraper, den Sie löschen möchten, und verwenden Sie dann die DeleteScraper-Operation, um ihn zu löschen. Alternativ mit dem AWS CLI, rufen Sie an:

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

Von Amazon gesammelte Kennzahlen OpenSearch Service

Wenn Sie mit Amazon integrieren OpenSearch Service, der Amazon Managed Service für Prometheus Collector, scrapft automatisch Prometheus-compatible Metriken, die den Zustand und die Leistung Ihrer Domain beschreiben. Der Collector gibt mehrere hundert Metriken aus. In den folgenden Tabellen sind repräsentative Metriken aus jeder Kategorie aufgeführt. Die `opensearch_indices_` Die Metriken in den folgenden Tabellen sind domänenübergreifend aggregiert, und viele haben reine Primärvarianten und Gesamtvarianten (Suffix `_primary` oder `_total`). Der Collector gibt auch die gleichen Statistiken pro Index mit einem Präfix aus `opensearch_index_stats_`, und Kennzahlen pro Shard mit Präfix `opensearch_indices_shards_`.

Cluster-Metriken

Metrik	Beschreibung/Zweck
opensearch_cluster_health_status	Integritätsstatus des Clusters (grün, gelb oder rot).
opensearch_cluster_health_number_of_nodes	Anzahl der Knoten im -Cluster.
opensearch_cluster_health_number_of_ data_nodes	Anzahl der Datenknoten im Cluster.
opensearch_cluster_health_active_pri mary_shards	Anzahl der aktiven primären Shards.
opensearch_cluster_health_active_shards	Gesamtzahl der aktiven Shards, einschließlich primärer Shards und Replikat-Shards.
opensearch_cluster_health_relocating_shards	Anzahl der Shards, die zwischen Knoten verschoben werden.

Metrik	Beschreibung/Zweck
opensearch_cluster_health_initializing_shards	Anzahl der Shards, die initialisiert werden.
opensearch_cluster_health_unassigned_shards	Anzahl der Shards, die keinem Knoten zugewiesen sind.
opensearch_cluster_health_delayed_unassigned_shards	Anzahl der nicht zugewiesenen Shards, deren Zuweisung verzögert ist.
opensearch_cluster_health_number_of_pending_tasks	Anzahl der Aufgaben auf Clusterebene, die sich in der Warteschlange befinden und auf ihre Ausführung warten.
opensearch_cluster_health_number_of_in_flight_fetch	Anzahl der laufenden Shard-Abrufanfragen.
opensearch_cluster_health_task_max_waiting_in_queue_millis	Längste Zeit, die eine Aufgabe in der Warteschlange gewartet hat, in Millisekunden.

Knoten-Metriken

Metrik	Beschreibung/Zweck
opensearch_os_cpu_percent	Prozentsatz der vom Betriebssystem auf dem Knoten verwendeten CPU.
opensearch_os_load1	Durchschnittliche Auslastung des Betriebssystems in der letzten 1 Minute. Der Kollektor emittiert außerdem `opensearch_os_load5` und `opensearch_os_load15` .
opensearch_os_mem_used_bytes	Menge des verwendeten physischen Speichers in Byte. Der Kollektor sendet außerdem `opensearch_os_mem_free_bytes` , `opensearch_os_mem_actual_used_bytes` , und `opensearch_os_mem_actual_free_bytes` .

Metrik	Beschreibung/Zweck
Öffnen Sie search_jvm_memory_used_bytes	Menge des verwendeten JVM-Speichers in Byte, aufgeschlüsselt nach Speicherbereich (Heap und Nicht-Heap). Zu den verwandten Metriken gehören `opensearch_jvm_memory_committed_bytes` und `opensearch_jvm_memory_max_bytes` .
opensearch_jvm_gc_collection_seconds_count	Gesamtzahl der JVM-Garbage-Collection-Ereignisse. Verwenden Sie mit `opensearch_jvm_gc_collection_seconds_sum` , die Gesamtzeit, die für die Müllabfuhr aufgewendet wurde.
opensearch_jvm_uptime_seconds	JVM-Verfügbarkeit in Sekunden.
öffnen Sie search_thread_pool_active_count	Anzahl der aktiven Threads in jedem Threadpool. Zu den verwandten Metriken gehören `opensearch_thread_pool_queue_count` , `opensearch_thread_pool_rejected_count` , und `opensearch_thread_pool_completed_count` .
opensearch_filesystem_data_available_bytes	Menge an Speicherplatz, der dem Knoten zur Verfügung steht, in Byte. Zu den verwandten Metriken gehören `opensearch_filesystem_data_free_bytes` und `opensearch_filesystem_data_size_bytes` .
opensearch_filesystem_io_stats_device_read_operations_count	Anzahl der Lesevorgänge auf der Festplatte. Der Kollektor gibt auch die Anzahl der Schreib- und Gesamtvorgänge sowie die Lese- und Schreibgrößen aus (z. B. `opensearch_filesystem_io_stats_device_read_size_kilobytes_sum`).

Metrik	Beschreibung/Zweck
opensearch_process_cpu_percent	Prozentsatz der von der verwendeten CPU OpenSearch Prozess.
opensearch_process_mem_resident_size_bytes	Größe des residenten Speichers von OpenSearch Prozess, in Byte.
opensearch_process_open_files_count	Anzahl der Dateideskriptoren, die von der geöffnet wurden OpenSearch prozess.
opensearch_breakers_tripped	Gesamtzahl der Auslösevorgänge, mit denen jeder Leistungsschalter ausgelöst wurde. Zu den verwandten Metriken gehören `opensearch_breakers_estimated_size_bytes` und `opensearch_breakers_limit_size_bytes` .
opensearch_transport_rx_size_bytes_total	Gesamtmenge der über die Transportschicht zwischen den Knoten empfangenen Daten, in Byte. Der Kollektor sendet auch `opensearch_transport_tx_size_bytes_total` und die Anzahl der Pakete.
opensearch_indexing_pressure_current_all_in_bytes	Aktueller Speicher, der von Indexierungsanfragen verbraucht wird, in Byte, mit `opensearch_indexing_pressure_limit_in_bytes` als konfiguriertes Limit.

Metriken indexieren

Metrik	Beschreibung/Zweck
opensearch_indices_docs	Anzahl der Dokumente. Zu den verwandten Metriken gehören `opensearch_indices_docs_deleted` , `opensearch_indices`

Metrik	Beschreibung/Zweck
	`_docs_primary` , und `opensearch_indices_docs_total` .
opensearch_indices_store_size_bytes	Gesamtgröße der Indizes auf der Festplatte, in Byte, mit `opensearch_indices_store_size_bytes_primary` und `opensearch_indices_store_size_bytes_total` Varianten.
opensearch_indices_indexing_index_total	Gesamtzahl der indizierten Dokumente. Verwenden Sie mit `opensearch_indices_indexing_index_time_seconds_total` zur Indizierung der Latenz.
opensearch_indices_indexing_ist_gedrosselt	Gibt an, ob die Indizierung derzeit gedrosselt ist, mit `opensearch_indices_indexing_throttle_time_seconds_total` für die gesamte gedrosselte Zeit.
opensearch_indices_search_query_total	Gesamtzahl der Suchanfragen. Verwenden Sie mit `opensearch_indices_search_query_time_seconds` für die Latenz bei Abfragen.
opensearch_indices_search_fetch_total	Gesamtzahl der Abrufvorgänge, mit `opensearch_indices_search_fetch_time_seconds` für die Latenz beim Abrufen.
opensearch_indices_get_total	Gesamtzahl der Get-Operationen, mit `opensearch_indices_get_time_seconds` für Get-Latenz.

Metrik	Beschreibung/Zweck
opensearch_indices_merges_total	Gesamtzahl der abgeschlossenen Segmentzusammenführungen. Zu den verwandten Metriken gehören `opensearch_indices_merges_current` und `opensearch_indices_merges_total_time_seconds_total`.
opensearch_indices_refresh_total	Gesamtzahl der Indexaktualisierungsvorgänge, mit `opensearch_indices_refresh_time_seconds_total` für die Aktualisierungszeit.
opensearch_indices_flush_total	Gesamtzahl der Index-Flush-Operationen, mit `opensearch_indices_flush_time_seconds` für die Spülzeit.
opensearch_indices_segments_count	Anzahl der Segmente. Der Collector gibt auch segmentspezifische Speichermetriken aus (z. B. `opensearch_indices_segments_memory_bytes` und `opensearch_indices_segment_terms_memory_total`).
opensearch_indices_translog_operations	Anzahl der Operationen im Transaktionslog, mit `opensearch_indices_translog_size_in_bytes` für seine Größe.
opensearch_indices_fielddata_memory_size_bytes	Vom Felddaten-Cache verwendeter Speicher, in Byte, mit `opensearch_indices_fielddata_evictions` für Räumungen.

Metrik	Beschreibung/Zweck
opensearch_indices_query_cache_memory_size_bytes	Vom Abfrage-Cache verwendeter Speicher in Byte. Zu den verwandten Metriken gehören `opensearch_indices_query_cache_count` und `opensearch_indices_query_cache_evictions` .
opensearch_indices_request_cache_memory_size_bytes	Vom Anforderungs-Cache verwendeter Speicher in Byte. Zu den verwandten Metriken gehören `opensearch_indices_request_cache_count` und `opensearch_indices_request_cache_evictions` .

Einschränkungen

Der Amazonas OpenSearch Die Serviceintegration mit Amazon Managed Service für Prometheus hat die folgenden Einschränkungen:

- Wird nur unterstützt für OpenSearch Servicedomänen, die Zugriff auf Amazon Virtual Private Cloud haben. Domains mit öffentlichem Zugriff werden nicht unterstützt.
- Ein Scraper sammelt Metriken von einem einzigen OpenSearch Service-Domäne. Um Metriken von mehr als einer Domain zu sammeln, erstellen Sie für jede Domain einen separaten Scraper.

Was sind Prometheus-compatible Metriken?

Um Prometheus-Metriken aus Ihren Anwendungen und Ihrer Infrastruktur für die Verwendung in Amazon Managed Service for Prometheus zu extrahieren, müssen sie instrumentiert und verfügbar gemacht werden Prometheus-compatible Metriken von Prometheus-compatible `/metrics` Endpunkte. Sie können Ihre eigenen Metriken implementieren, müssen es aber nicht. Kubernetes (einschließlich Amazon EKS) und viele andere Bibliotheken und Services implementieren diese Metriken direkt.

Wenn Metriken in Amazon EKS in eine exportiert werden Prometheus-compatible Endpunkt, Sie können diese Metriken automatisch vom Amazon Managed Service for Prometheus Collector löschen lassen.

Weitere Informationen finden Sie unter den folgenden Themen:

- Weitere Informationen zu vorhandenen Bibliotheken und Services, die Metriken als Prometheus-Metriken exportieren, finden Sie unter Exporter und Integrationen in der Prometheus-Dokumentation.
- Weitere Informationen zum Exportieren Prometheus-compatible Metriken aus Ihrem eigenen Code finden Sie unter Exporteure schreiben in der Prometheus-Dokumentation.
- Weitere Informationen darüber, wie Sie einen Kollektor für Amazon Managed Service für Prometheus einrichten, der automatisch Metriken aus Ihren Amazon-EKS-Clustern abruft, finden Sie unter Richten Sie verwaltete Collectors für Amazon EKS ein.

Überwachen Sie Sammler mit verkauften Baumstämmen

Die Sammler von Amazon Managed Service for Prometheus stellen veräußerte Protokolle zur Verfügung, mit denen Sie den Prozess der Erfassung von Kennzahlen überwachen und Fehler beheben können. Diese Protokolle werden automatisch an Amazon CloudWatch Logs gesendet und bieten Einblick in die Serviceerkennung, die Erfassung von Metriken und den Datenexport. Der Collector verkauft Protokolle für drei Hauptkomponenten der Pipeline zur Erfassung von Metriken:

Themen

- Protokolle zur Diensterkennung
- Collector-Protokolle
- Exporter-Protokolle
- Collector Vended Logs verstehen und verwenden

Protokolle zur Diensterkennung

Service-Discovery-Protokolle enthalten Informationen über den Zielermittlungsprozess, darunter:

- Authentifizierungs- oder Berechtigungsprobleme beim Zugriff auf Kubernetes-API-Ressourcen.
- Konfigurationsfehler in den Service Discovery-Einstellungen.

Die folgenden Beispiele zeigen häufige Authentifizierungs- und Berechtigungsfehler, die bei der Serviceerkennung auftreten können:

Amazon EKS-Cluster nicht vorhanden

Wenn der angegebene Amazon EKS-Cluster nicht existiert, erhalten Sie die folgende Fehlermeldung:

```
{
  "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"
}
```

Ungültige Berechtigungen für Dienste

Wenn der Collector nicht über die erforderlichen RBAC-Berechtigungen (Role-Based Access Control) verfügt, um Dienste zu überwachen, wird folgende Fehlermeldung angezeigt:

```
{
  "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"
}
```

Ungültige Berechtigungen für Endgeräte

Wenn der Collector nicht über die erforderlichen RBAC-Berechtigungen (Role-Based Access Control) verfügt, um Endpoints zu überwachen, wird folgende Fehlermeldung angezeigt:

```
{
  "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"
}
```

Collector-Protokolle

Collector-Protokolle enthalten Informationen über den Prozess des Auslesens von Metriken, darunter:

- Scrape-Fehler, die darauf zurückzuführen sind, dass Endgeräte nicht verfügbar sind.
- Verbindungsprobleme beim Versuch, Ziele zu scrapen.
- Timeouts bei Scrape-Vorgängen.
- Von Scrape-Zielen zurückgegebene HTTP-Statusfehler.

Die folgenden Beispiele zeigen häufige Collector-Fehler, auf die Sie beim Scraping von Metriken stoßen können:

Endpunkt fehlender Metriken

Wenn der `/metrics` Endpunkt auf der Zielinstanz nicht verfügbar ist, erhalten Sie diesen Fehler:

```
{
    "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"
}
```

Verbindung wurde verweigert

Wenn der Collector keine Verbindung zum Zielendpunkt herstellen kann, erhalten Sie diesen Fehler:

```
{
  "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"
}
```

Exporter-Protokolle

Exportprotokolle enthalten Informationen über den Prozess des Sendens der gesammelten Messwerte an Ihren Amazon Managed Service for Prometheus-Workspace, einschließlich:

- Anzahl der verarbeiteten Metriken und Datenpunkte.
- Exportfehler aufgrund von Workspace-Problemen.
- Berechtigungsfehler beim Versuch, Metriken zu schreiben.
- Abhängigkeitsfehler in der Exportpipeline.

Das folgende Beispiel zeigt einen häufigen Exportfehler, der beim Export von Metriken auftreten kann:

Workspace wurde nicht gefunden

Wenn der Ziel-Workspace für den Metrikexport nicht gefunden werden kann, wird folgende Fehlermeldung angezeigt:

```
{
    "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"
}
```

Collector Vended Logs verstehen und verwenden

Struktur der Protokolle

Alle von Collector verkauften Logs folgen einer konsistenten Struktur mit den folgenden Feldern:

scrapeConfigId

Die eindeutige Kennung der Scrape-Konfiguration, die das Protokoll generiert hat.

timestamp

Der Zeitpunkt, zu dem der Protokolleintrag generiert wurde.

message

Der Inhalt der Protokollnachricht, der zusätzliche strukturierte Felder enthalten kann.

Komponente

Die Komponente, die das Protokoll generiert hat (SERVICE_DISCOVERY, COLLECTOR oder EXPORTER)

Verwendung von versendeten Protokollen zur Fehlerbehebung

Die Collector Vended Logs helfen Ihnen bei der Behebung häufiger Probleme bei der Erfassung von Metriken:

1. Probleme bei der Serviceerkennung
   - Überprüfen Sie die SERVICE_DISCOVERY-Protokolle auf Authentifizierungs- oder Berechtigungsfehler.
   - Stellen Sie sicher, dass der Collector über die erforderlichen Berechtigungen für den Zugriff auf Kubernetes-Ressourcen verfügt.
2. Probleme beim Scraping von Metriken
   - Überprüfen Sie die COLLECTOR-Protokolle auf Scrape-Fehler.
   - Stellen Sie sicher, dass auf die Zielendpunkte zugegriffen werden kann und ob Metriken zurückgegeben werden.
   - Stellen Sie sicher, dass die Firewallregeln es dem Collector ermöglichen, eine Verbindung zu Zielendpunkten herzustellen.
3. Probleme beim Export von Metriken
   - Überprüfen Sie die EXPORTER-Protokolle auf Exportfehler.

- Stellen Sie sicher, dass der Workspace existiert und korrekt konfiguriert ist.
- Stellen Sie sicher, dass der Collector über die erforderlichen Berechtigungen verfügt, um in den Workspace zu schreiben.

Zugreifen auf vom Collector versendete Protokolle

Von Collector verkaufte Logs werden automatisch an Amazon CloudWatch Logs gesendet. So greifen Sie auf diese Protokolle zu:

1. Öffnen Sie die CloudWatch Konsole unter https://console.aws.amazon.com/cloudwatch/.
2. Wählen Sie im Navigationsbereich Protokollgruppen aus.
3. Suchen Sie die Protokollgruppe für Ihren Collector und wählen Sie sie aus:`/aws/prometheus/workspace_id/collector/collector_id`.
4. Durchsuchen oder durchsuchen Sie die Protokollereignisse, um relevante Informationen zu finden.

Sie können CloudWatch Logs Insights auch verwenden, um Ihre Collector-Logs abzufragen und zu analysieren. Um beispielsweise alle Fehler bei der Diensterkennung zu finden, gehen Sie wie folgt vor:

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

Bewährte Methoden für die Überwachung von Kollektoren

So überwachen Sie Ihren Amazon Managed Service für Prometheus-Sammler effektiv:

1. Richten Sie CloudWatch Alarme für kritische Collector-Probleme ein, wie z. B. anhaltende Scrape-Ausfälle oder Exportfehler. Weitere Informationen finden Sie unter Alarme im CloudWatch Amazon-Benutzerhandbuch.
2. Erstellen Sie CloudWatch Dashboards, um die Leistungskennzahlen von Collector zusammen mit den gesammelten Protokolldaten zu visualisieren. Weitere Informationen finden Sie unter Dashboards im CloudWatch Amazon-Benutzerhandbuch.
3. Überprüfen Sie regelmäßig die Service-Discovery-Logs, um sicherzustellen, dass die Ziele korrekt erkannt werden.

4. Überwachen Sie die Anzahl der gelöschten Ziele, um potenzielle Konfigurationsprobleme zu identifizieren.
5. Verfolgen Sie Exportfehler, um sicherzustellen, dass die Metriken erfolgreich an Ihren Workspace gesendet werden.

Kundenverwaltete Kollektoren

Dieser Abschnitt enthält Informationen zum Erfassen von Daten, indem Sie Ihre eigenen Kollektoren einrichten, die mithilfe von Prometheus Remote-Write-Metriken an Amazon Managed Service für Prometheus senden.

Wenn Sie Ihre eigenen Kollektoren verwenden, um Metriken an Amazon Managed Service für Prometheus zu senden, sind Sie dafür verantwortlich, Ihre Metriken zu sichern und sicherzustellen, dass der Erfassungsprozess Ihren Verfügbarkeitsanforderungen entspricht.

Die meisten vom Kunden verwalteten Kollektoren verwenden eines der folgenden Tools:

- AWS Distro for OpenTelemetry (ADOT) — ADOT ist eine vollständig unterstützte, sichere und produktionsbereite Open-Source-Distribution, die Agenten zur Erfassung von OpenTelemetry Kennzahlen bereitstellt. Sie können ADOT verwenden, um Metriken zu sammeln und sie an Ihren Workspace in Amazon Managed Service für Prometheus zu senden. Weitere Informationen zum ADOT Collector finden Sie unter Distro for. AWS OpenTelemetry
- Prometheus-Agent – Sie können Ihre eigene Instance des Open-Source-Prometheus-Servers einrichten, der als Agent ausgeführt wird, um Metriken zu sammeln und sie an Ihren Workspace in Amazon Managed Service für Prometheus weiterzuleiten.

Die folgenden Themen beschreiben die Verwendung dieser beiden Tools und enthalten allgemeine Informationen zur Einrichtung Ihrer eigenen Kollektoren.

Themen

- Sichern Sie die Erfassung Ihrer Metriken
- Verwenden AWS Distro für OpenTelemetry als Sammler
- Eine Prometheus-Instance als Kollektor verwenden
- Richten Sie Amazon Managed Service für Prometheus für Daten mit hoher Verfügbarkeit ein

Sichern Sie die Erfassung Ihrer Metriken

Amazon Managed Service für Prometheus bietet Möglichkeiten, Sie bei der sicheren Erfassung Ihrer Messwerte zu unterstützen.

Verwenden AWS PrivateLink mit Amazon Managed Service für Prometheus

Der Netzwerkverkehr zur Erfassung der Metriken in Amazon Managed Service for Prometheus kann über einen öffentlichen Internetendpunkt oder über einen VPC-Endpunkt erfolgen. AWS PrivateLink Durch die Verwendung von AWS PrivateLink wird sichergestellt, dass der Netzwerkverkehr von Ihren VPCs innerhalb des AWS Netzwerks gesichert ist, ohne dass er über das öffentliche Internet übertragen wird. Informationen zum Erstellen eines AWS PrivateLink VPC-Endpunkts für Amazon Managed Service for Prometheus finden Sie unter. Verwendung von Amazon Managed Service für Prometheus mit Schnittstellen-VPC-Endpunkten

Authentifizierung und Autorisierung

AWS Identity and Access Management (IAM) ist ein Webservice, mit dem Sie den Zugriff auf Ressourcen sicher kontrollieren können. AWS Sie verwenden IAM, um zu steuern, wer authentifiziert (angemeldet) und autorisiert (Berechtigungen besitzt) ist, Ressourcen zu nutzen. Amazon Managed Service für Prometheus ist in IAM integriert, um Ihnen zu helfen, Ihre Daten zu schützen. Wenn Sie Amazon Managed Service für Prometheus einrichten, müssen Sie einige IAM-Rollen erstellen, die es ermöglichen, Metriken von Prometheus-Servern zu erfassen, und die es Grafana-Servern ermöglichen, die Metriken abzufragen, die in Ihren Workspace in Amazon Managed Service für Prometheus gespeichert sind. Weitere Informationen zu IAM finden Sie unter Was ist IAM?.

Eine weitere AWS Sicherheitsfunktion, die Ihnen bei der Einrichtung von Amazon Managed Service für Prometheus helfen kann, ist der AWS Signature Version 4-Signaturprozess (AWS Sigv4). Signature Version 4 ist der Prozess zum Hinzufügen von Authentifizierungsinformationen zu AWS Anfragen, die über HTTP gesendet werden. Aus Sicherheitsgründen AWS müssen die meisten Anfragen mit einem Zugriffsschlüssel signiert werden, der aus einer Zugriffsschlüssel-ID und einem geheimen Zugriffsschlüssel besteht. Diese beiden Schlüssel werden in der Regel als Sicherheitsanmeldeinformationen bezeichnet. Weitere Informationen über SigV4 finden Sie unter Signatur Version 4-Signierungsprozess.

Verwenden AWS Distro für OpenTelemetry als Sammler

In diesem Abschnitt wird beschrieben, wie Sie den AWS Distro for OpenTelemetry (ADOT) Collector so konfigurieren, dass er Daten aus einer Prometheus-instrumented Anwendung scrapt und die

Metriken an Amazon Managed Service for Prometheus sendet. Weitere Informationen zum ADOT Collector finden Sie unter Distro für. AWS OpenTelemetry

In den folgenden Themen werden drei verschiedene Möglichkeiten beschrieben, ADOT als Collector für Ihre Metriken einzurichten, je nachdem, ob Ihre Metriken von Amazon EKS, Amazon ECS oder einer Amazon EC2-Instance stammen.

Themen

- Richten Sie die Erfassung von Metriken ein mit AWS Distribution für OpenTelemetry auf einem Amazon Elastic Kubernetes Service-Cluster
- Richten Sie die Erfassung von Metriken aus Amazon ECS mit ein AWS Distribution für Open Telemetry
- Einrichten der Erfassung von Metriken aus einer Amazon-EC2-Instance mithilfe von Remote-Write

Richten Sie die Erfassung von Metriken ein mit AWS Distribution für OpenTelemetry auf einem Amazon Elastic Kubernetes Service-Cluster

Sie können den AWS Distro for OpenTelemetry (ADOT) Collector verwenden, um Metriken aus einer Prometheus-instrumented Anwendung zu entfernen und die Metriken an Amazon Managed Service for Prometheus zu senden.

> **Note**
>
> Weitere Informationen zum ADOT-Collector finden Sie unter Distro für. AWS OpenTelemetry Weitere Informationen zu Prometheus-instrumented Anwendungen finden Sie unter. Was sind Prometheus-compatible Metriken?

Das Erfassen von Prometheus-Metriken mit ADOT umfasst drei OpenTelemetry Komponenten: den Prometheus-Empfänger, den Prometheus Remote Write Exporter und die Sigv4-Authentifizierungserweiterung.

Sie können den Prometheus-Empfänger mithilfe Ihrer vorhandenen Prometheus-Konfiguration konfigurieren, um die Serviceerkennung und das Metrik-Scraping durchzuführen. Der Prometheus Receiver erfasst Metriken im Prometheus-Expositionsformat. Alle Anwendungen oder Endpunkte, die Sie scrapen möchten, sollten mit der Prometheus-Client-Bibliothek konfiguriert werden. Der Prometheus Receiver unterstützt alle Prometheus-Konfigurationen zum Scraping und Umetikettieren,

die in der Prometheus-Dokumentation unter Konfiguration beschrieben sind. Sie können diese Konfigurationen direkt in Ihre ADOT Kollektor-Konfigurationen einfügen.

Der Prometheus Remote Write Exporter verwendet den `remote_write`-Endpunkt, um die gesammelten Metriken an Ihren Management-Portal-Workspace zu senden. Die HTTP-Anfragen zum Exportieren von Daten werden mit SigV4, dem AWS Protokoll für sichere Authentifizierung, mit der AWS Sigv4-Authentifizierungserweiterung signiert. Weitere Informationen finden Sie unter Signatur Version 4-Signierungsprozess.

Der Kollektor erkennt automatisch Prometheus-Metriken-Endpunkte auf Amazon EKS und verwendet die Konfiguration unter <kubernetes_sd_config>.

Die folgende Demo ist ein Beispiel für diese Konfiguration auf einem Cluster, auf dem Amazon Elastic Kubernetes Service oder selbstverwaltetes Kubernetes ausgeführt wird. Um diese Schritte ausführen zu können, benötigen Sie AWS Anmeldeinformationen aus einer der möglichen Optionen in der standardmäßigen AWS Anmeldeinformationskette. Weitere Informationen finden Sie unter Konfiguration des AWS SDK für Go. In dieser Demo wird eine Beispiel-App genutzt, die für Integrationstests des Prozesses verwendet wird. Die Beispiel-App stellt Metriken am `/metrics`-Endpunkt bereit, wie die Prometheus-Client-Bibliothek.

Voraussetzungen

Bevor Sie mit den folgenden Schritten zur Einrichtung der Datenerfassung beginnen, müssen Sie Ihre IAM-Rolle für das Servicekonto und die Vertrauensrichtlinie einrichten.

So richten Sie die IAM-Rolle für das Servicekonto und die Vertrauensrichtlinie ein

1. Erstellen Sie die IAM-Rolle für das Servicekonto, indem Sie die Schritte unter Einrichten von Servicerollen für die Erfassung von Metriken aus Amazon-EKS-Clustern befolgen.

   Der ADOT Kollektor verwendet diese Rolle, wenn er Metriken erfasst und exportiert.
2. Bearbeiten Sie als Nächstes die Vertrauensrichtlinie. Öffnen Sie unter https://console.aws.amazon.com/iam/ die IAM-Konsole.
3. Wählen Sie im linken Navigationsbereich Rollen aus und suchen Sie nach der amp-iamproxy-ingest-role, die Sie in Schritt 1 erstellt haben.
4. Wählen Sie die Registerkarte Vertrauensbeziehungen und dann Vertrauensbeziehung bearbeiten aus.

5. Ersetzen Sie in der JSON-Vertrauenbeziehungsrichtlinie `aws-amp` durch `adot-col` und wählen Sie dann Vertrauensrichtlinie aktualisieren aus. Das Ergebnis Ihrer Vertrauensbeziehungsrichtlinie sollte wie folgt aussehen:

   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. Wählen Sie die Registerkarte Berechtigungen und stellen Sie sicher, dass die folgende Berechtigungsrichtlinie an die Rolle angehängt ist.

   JSON

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

```
                "aps:GetSeries",
                "aps:GetLabels",
                "aps:GetMetricMetadata"
            ],
            "Resource": "*"
        }
    ]
}
```

Erfassung der Prometheus-Metriken aktivieren

> **Note**
>
> Wenn Sie einen Namespace in Amazon EKS erstellen, sind `alertmanager` und Node Exporter standardmäßig deaktiviert.

So aktivieren Sie die Prometheus-Erfassung auf einem Amazon-EKS- oder Kubernetes-Cluster

1. Forken und klonen Sie die Beispiel-App aus dem Repository auf der aws-otel-community.

   Führen Sie anschließend folgende Befehle aus.

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

2. Schieben Sie dieses Bild in eine Registrierung wie Amazon ECR oder DockerHub.
3. Stellen Sie die Beispiel-App im Cluster bereit, indem Sie diese Kubernetes-Konfiguration kopieren und anwenden. Ändern Sie das Bild in das Bild, das Sie gerade übertragen haben, indem Sie `{{PUBLIC_SAMPLE_APP_IMAGE}}` in der `prometheus-sample-app.yaml` Datei ersetzen.

   ```
   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. Geben Sie den folgenden Befehl ein, um zu überprüfen, ob die Beispiel-App gestartet wurde. In der Ausgabe des Befehls sehen Sie `prometheus-sample-app` in der `NAME` Spalte.

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

5. Starten Sie eine Standard-Instance des ADOT Kollektors. Geben Sie dazu zunächst den folgenden Befehl ein, um die Kubernetes-Konfiguration für ADOT Kollektor abzurufen.

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

   Bearbeiten Sie dann die Vorlagendatei und ersetzen Sie den remote_write-Endpunkt für Ihren Workspace in Amazon Managed Service für Prometheus durch `YOUR_ENDPOINT` und Ihre Region durch `YOUR_REGION`. Verwenden Sie den remote_write-Endpunkt, der in der Konsole von Amazon Managed Service für Prometheus angezeigt wird, wenn Sie sich Ihre Workspace-Details ansehen.

   Sie müssen auch `YOUR_ACCOUNT_ID` im Abschnitt „Dienstkonto“ der Kubernetes-Konfiguration Ihre AWS Konto-ID ändern.

   In diesem Beispiel verwendet die ADOT Kollektor-Konfiguration eine Anmerkung (`scrape=true`), um anzugeben, welche Zielendpunkte gescrapt werden sollen. Auf diese Weise kann der ADOT Kollektor den Endpunkt der Beispiel-App von den Endpunkten des Kube-Systems in Ihrem Cluster unterscheiden. Sie können dies aus den Konfigurationen der Umetikettierung entfernen, wenn Sie eine andere Beispiel-App entfernen möchten.

6. Geben Sie den folgenden Befehl ein, um den ADOT Kollektor bereitzustellen.

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

7. Geben Sie den folgenden Befehl ein, um zu überprüfen, ob der ADOT Kollektor gestartet wurde. Suchen Sie nach `adot-col` in der `NAMESPACE` Spalte.

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

8. Stellen Sie mithilfe des Protokoll-Exporters sicher, dass die Pipeline funktioniert. Unsere Beispielvorlage ist bereits in den Protokoll-Exporter integriert. Geben Sie die folgenden Befehle ein.

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

Einige der aus der Beispiel-App gewonnenen Metriken sehen wie im folgenden Beispiel aus.

```
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. Um zu testen, ob Amazon Managed Service für Prometheus die Messwerte erhalten hat, verwenden Sie `awscurl`. Mit diesem Tool können Sie HTTP-Anfragen mit AWS Sigv4-Authentifizierung über die Befehlszeile senden. Sie müssen also über lokal eingerichtete AWS Anmeldeinformationen mit den richtigen Berechtigungen verfügen, um Abfragen von Amazon Managed Service for Prometheus durchführen zu können. Anweisungen zur Installation finden Sie unter awscurl. `awscurl` https://github.com/okigan/awscurl

   Ersetzen Sie `AMP_REGION` im folgenden Befehl und `AMP_ENDPOINT` durch die Informationen für Ihren .

   ```
   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"]}]}}
   ```

   Wenn Sie als Antwort eine Metrik erhalten, bedeutet das, dass Ihre Pipeline-Einrichtung erfolgreich war und die Metrik erfolgreich von der Beispiel-App an Amazon Managed Service für Prometheus weitergegeben wurde.

Bereinigen

Geben Sie die folgenden Befehle ein, um diese Demo zu bereinigen.

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

Erweiterte Konfiguration

Der Prometheus Receiver unterstützt alle Prometheus-Konfigurationen zum Scraping und Umetikettieren, die in der Prometheus-Dokumentation unter Konfiguration beschrieben sind. Sie können diese Konfigurationen direkt in Ihre ADOT Kollektor-Konfigurationen einfügen.

Die Konfiguration für den Prometheus Receiver umfasst Ihre Serviceerkennung, Scraping-Konfigurationen und Umetikettierungs-Konfigurationen. Die Konfiguration des Empfängers sieht wie folgt aus.

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

Folgendes ist ein Konfigurationsbeispiel:

```
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
```

Wenn Sie über eine bestehende Prometheus-Konfiguration verfügen, müssen Sie die $ Zeichen durch $$ ersetzen, um zu vermeiden, dass die Werte durch Umgebungsvariablen ersetzt werden. *

Dies ist besonders wichtig für den Ersatzwert der relabel_configurations. Wenn Sie beispielsweise mit der folgenden relabel_configuration beginnen:

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

Es würde wie folgt werden:

```
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 und SigV4-Authentifizierungserweiterung

Die Konfiguration für den Prometheus Remote Write Exporter und die SigV4-Authentifizierungserweiterung ist einfacher als für den Prometheus-Empfänger. In dieser Phase der Pipeline wurden die Metriken bereits erfasst und wir sind bereit, diese Daten in Amazon Managed Service für Prometheus zu exportieren. Die Mindestanforderung für eine erfolgreiche Konfiguration für die Kommunikation mit Amazon Managed Service für Prometheus ist im folgenden Beispiel dargestellt.

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

Diese Konfiguration sendet eine HTTPS-Anfrage, die von AWS Sigv4 mit AWS Anmeldeinformationen aus der Standard-Anmeldeinformationskette signiert wird. AWS Weitere

Informationen finden Sie unter Konfigurieren der AWS SDK für Go. Sie müssen den Service als `aps` angeben.

Unabhängig von der Bereitstellungsmethode muss der ADOT-Collector Zugriff auf eine der aufgelisteten Optionen in der AWS Standard-Anmeldeinformationskette haben. Die Sigv4-Authentifizierungserweiterung hängt von der ab AWS SDK für Go und verwendet sie zum Abrufen von Anmeldeinformationen und zur Authentifizierung. Sie müssen sicherstellen, dass diese Anmeldeinformationen über Remote-Write-Berechtigungen für Amazon Managed Service für Prometheus verfügen.

Richten Sie die Erfassung von Metriken aus Amazon ECS mit ein AWS Distribution für Open Telemetry

In diesem Abschnitt wird erklärt, wie Sie Metriken von Amazon Elastic Container Service (Amazon ECS) sammeln und sie mithilfe von AWS Distro for Open Telemetry (ADOT) in Amazon Managed Service for Prometheus importieren. Außerdem wird beschrieben, wie Sie Ihre Metriken in Amazon Managed Grafana anzeigen können.

Voraussetzungen

> **Important**
>
> Bevor Sie beginnen, benötigen Sie eine Amazon-ECS-Umgebung auf einem AWS Fargate -Cluster mit Standardeinstellungen, einen Workspace in Amazon Managed Service für Prometheus und einen Workspace in Amazon Managed Grafana. Wir gehen davon aus, dass Sie mit Container-Workloads, Amazon Managed Service für Prometheus und Amazon Managed Grafana vertraut sind.

Weitere Informationen finden Sie unter den folgenden Links:

- Informationen zum Erstellen einer Amazon ECS-Umgebung auf einem Fargate-Cluster mit Standardeinstellungen finden Sie unter Erstellen eines Clusters im Amazon ECS-Entwicklerhandbuch.
- Informationen zum Erstellen eines Workspace in Amazon Managed Service für Prometheus finden Sie unter Einen Workspace erstellen im Benutzerhandbuch von Amazon Managed Service für Prometheus.
- Informationen zum Erstellen eines Workspace in Amazon Managed Grafana-finden Sie unter Einen Workspace erstellen im Benutzerhandbuch von Amazon Managed Grafana.

Schritt 1: Definieren Sie ein benutzerdefiniertes ADOT-Collector-Container-Image

Verwenden Sie die folgende Konfigurationsdatei als Vorlage, um Ihr eigenes ADOT-Kollektor-Container-Image zu definieren. Ersetzen Sie *my-remote-URL* und *my-region* durch Ihre `endpoint` und `region` -Werte. Speichern Sie die Konfiguration in einer Datei namens adot-config.yaml.

> **Note**
>
> Diese Konfiguration verwendet die `sigv4auth` Erweiterung zur Authentifizierung von Aufrufen an Amazon Managed Service für Prometheus. Weitere Informationen zur Konfiguration `sigv4auth` finden Sie unter Authenticator — Sigv4 on. 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]
```

Schritt 2: Übertragen Sie Ihr ADOT-Collector-Container-Image in ein Amazon ECR-Repository

Verwenden Sie ein Dockerfile, um Ihr Container-Image zu erstellen und in ein Repository von Amazon Elastic Container Registry (ECR) zu pushen.

1. Erstellen Sie das Dockerfile, um Ihr Container-Image zu kopieren und dem OTEL Docker-Image hinzuzufügen.

   ```
   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. Erstellen Sie ein Amazon-ECR-Repository.

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

3. Erstellen Sie Ihr Container-Image.

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

> **Note**
>
> Dies setzt voraus, dass Sie Ihren Container in derselben Umgebung erstellen, in der er ausgeführt werden soll. Wenn nicht, müssen Sie den `--platform` Parameter möglicherweise beim Erstellen des Images verwenden.

4. Melden Sie sich im Amazon-ECR-Repository an. Ersetzen Sie durch *my-region* Ihren Wert`region`.

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

5. Pushen Sie Ihr Container-Image.

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

Schritt 3: Erstellen Sie eine Amazon ECS-Aufgabendefinition, um Amazon Managed Service für Prometheus zu durchsuchen

Erstellen Sie eine Amazon-ECS-Aufgabendefinition zum Scrapen von Amazon Managed Service für Prometheus. Ihre Aufgabendefinition sollte einen Container mit dem Namen `adot-collector` und einen Container mit dem Namen `prometheus` enthalten. `prometheus` generiert Metriken und `adot-collector` Scrapes`prometheus`.

> **Note**
>
> Amazon Managed Service für Prometheus wird als Service ausgeführt und sammelt Metriken aus Containern. In diesem Fall führen die Container Prometheus lokal im Agentenmodus aus, wodurch die lokalen Metriken an Amazon Managed Service für Prometheus gesendet werden.

Beispiel: Aufgabendefinition

Im Folgenden finden Sie ein Beispiel dafür, wie Ihre Aufgabendefinition aussehen kann. Sie können dieses Beispiel als Vorlage verwenden, um Ihre eigene Aufgabendefinition zu erstellen. Ersetzen Sie den `image`-Wert von `adot-collector` durch Ihre Repository-URL und Ihr Bild-Tag (`$COLLECTOR_REPOSITORY:ecs`). Ersetzen Sie die `region`-Werte von `adot-collector` und `prometheus` durch Ihre `region`-Werte.

```
{
  "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"
}
```

Schritt 4: Erteilen Sie Ihrer Aufgabe die Berechtigungen für den Zugriff auf Amazon Managed Service for Prometheus

Um die gescrapten Metriken an Amazon Managed Service for Prometheus zu senden, muss Ihre Amazon ECS-Aufgabe über die richtigen Berechtigungen verfügen, um die API-Operationen für Sie aufrufen zu können. AWS Sie müssen eine IAM-Rolle für Ihre Aufgaben erstellen und die `AmazonPrometheusRemoteWriteAccess`-Richtlinie anhängen. Weitere Informationen zum Erstellen dieser Rolle und Anhängen der Richtlinie finden Sie unter Eine IAM-Rolle und Richtlinie für Ihre Aufgaben erstellen.

Nachdem Sie `AmazonPrometheusRemoteWriteAccess` zu Ihrer IAM-Rolle hinzufügen und diese Rolle für Ihre Aufgaben verwendet haben, kann Amazon ECS Ihre gescrapten Metriken an Amazon Managed Service für Prometheus senden.

Schritt 5: Visualisieren Sie Ihre Metriken in Amazon Managed Grafana

> **Important**
>
> Bevor Sie beginnen, müssen Sie eine Fargate-Aufgabe in Ihrer Amazon-ECS-Aufgabendefinition ausführen. Andernfalls kann Amazon Managed Service für Prometheus Ihre Metriken nicht verarbeiten.

1. Wählen Sie im Navigationsbereich Ihres Amazon Managed Grafana-Arbeitsbereichs unter dem AWS Symbol Datenquellen aus.
2. Wählen Sie auf der Registerkarte Datenquellen unter Service die Option Amazon Managed Service für Prometheus und Ihre Standardregion aus.
3. Wählen Sie Datenquelle hinzufügen aus.
4. Verwenden Sie die Präfixe `ecs` und `prometheus`, um Ihre Metriken abzufragen und anzuzeigen.

Einrichten der Erfassung von Metriken aus einer Amazon-EC2-Instance mithilfe von Remote-Write

In diesem Abschnitt wird erläutert, wie Sie einen Prometheus-Server mit Remote-Write in einer Instance von Amazon Elastic Compute Cloud (Amazon EC2) ausführen. Es wird erklärt, wie Sie Metriken aus einer in Go geschriebenen Demo-Anwendung sammeln und sie an einen Workspace in Amazon Managed Service für Prometheus senden.

Voraussetzungen

> **Important**
>
> Bevor Sie beginnen, müssen Sie Prometheus v2.26 oder höher installiert haben. Wir gehen davon aus, dass Sie mit Prometheus, Amazon EC2 und Amazon Managed Service für Prometheus vertraut sind. Informationen zur Installation von Prometheus finden Sie unter Erste Schritte auf der Prometheus-Website.

Wenn Sie mit Amazon EC2 oder Amazon Managed Service für Prometheus nicht vertraut sind, empfehlen wir Ihnen, zunächst die folgenden Abschnitte zu lesen:

- Was ist Amazon Elastic Compute Cloud?
- Was ist Amazon Managed Service für Prometheus?

Erstellen Sie eine IAM-Rolle für Amazon EC2

Um Metriken zu streamen, müssen Sie zunächst eine IAM-Rolle mit der AWS verwalteten Richtlinie erstellen. AmazonPrometheusRemoteWriteAccess Anschließend können Sie eine Instance mit der Rolle starten und Metriken in Ihren Workspace in Amazon Managed Service für Prometheus streamen.

1. Öffnen Sie unter https://console.aws.amazon.com/iam/ die IAM-Konsole.
2. Wählen Sie im Navigationsbereich Roles (Rollen) und dann Create role (Rolle erstellen) aus.
3. Wählen Sie als Typ der vertrauenswürdigen Entität AWS -Service aus. Wählen Sie für den Anwendungsfall EC2 aus. Wählen Sie Weiter: Berechtigungen aus.
4. Geben Sie im Suchfeld AmazonPrometheusRemoteWriteAccess ein. Wählen Sie AmazonPrometheusRemoteWriteAccess als Richtlinienname die Option und dann Richtlinie anhängen aus. Wählen Sie Next:Tags.

5. (Optional) Erstellen Sie IAM-Tags für Ihre IAM-Rolle. Wählen Sie Weiter: Prüfen aus.
6. Geben Sie einen Namen für die neue Rolle ein. Wählen Sie Richtlinie erstellen aus.

Starten Sie eine Amazon-EC2-Instance

Befolgen Sie die Anweisungen unter Eine Instance starten im Amazon EC2-Benutzerhandbuch für Linux-Instances, um eine Amazon-EC2-Instance zu starten.

Setzen Sie die Demo-App ein

Nachdem Sie Ihre IAM-Rolle erstellt und eine EC2-Instance mit dieser Rolle gestartet haben, können Sie eine Demoanwendung ausführen, um zu sehen, wie sie funktioniert.

Um eine Demoanwendung auszuführen und Metriken zu testen

1. Verwenden Sie die folgende Vorlage, um eine Go-Datei mit dem Namen `main.go` zu erstellen.

```
package main

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

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

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

2. Führen Sie den folgenden Befehl aus, um die richtigen Abhängigkeiten zu installieren.

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

3. Setzen Sie die Demo-App ein.

```
go run main.go
```

Die Demo-App sollte auf Port 8000 laufen und alle offengelegten Prometheus-Metriken anzeigen. Das ist ein Beispiel dieser Metriken.

```
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
```

Erstellen eines Workspace in Amazon Managed Service für Prometheus

Um einen Workspace in Amazon Managed Service für Prometheus zu erstellen, folgen Sie den Anweisungen unter Einen Workspace erstellen.

Setzen Sie einen Prometheus-Server ein

1. Verwenden Sie die folgende YAML-Beispieldatei als Vorlage, um eine neue Datei mit dem Namen `prometheus.yaml` zu erstellen. *`my-region`*Ersetzen Sie das durch Ihren Regionswert und *`my-workspace-id`* durch die Workspace-ID, die Amazon Managed Service for Prometheus für Sie generiert hat. `url` Für`region`, ersetze es *`my-region`* durch deinen Regionswert.

   Beispiel: YAML-Datei

```
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. Führen Sie den Prometheus-Server aus, um die Metriken der Demo-App an Ihren Workspace in Amazon Managed Service für Prometheus zu senden.

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

Der Prometheus-Server sollte jetzt die Metriken der Demo-App an Ihren Workspace in Amazon Managed Service für Prometheus senden.

Eine Prometheus-Instance als Kollektor verwenden

Sie können eine Prometheus-Instance verwenden, die im Agentenmodus ausgeführt wird (bekannt als Prometheus-Agent), um Metriken zu scrapen und sie an Ihren Amazon Managed Service for Prometheus-Workspace zu senden.

In den folgenden Themen werden verschiedene Möglichkeiten beschrieben, eine im Agentenmodus ausgeführte Prometheus-Instance als Kollektor für Ihre Metriken einzurichten.

> **Warning**
>
> Wenn Sie einen Prometheus-Agenten erstellen, sind Sie für dessen Konfiguration und Wartung verantwortlich. Vermeiden Sie es, Prometheus-Scrape-Endpunkte dem öffentlichen Internet auszusetzen, indem Sie Sicherheitsfunktionen aktivieren. https://prometheus.io/docs/prometheus/latest/configuration/https/

Wenn Sie mehrere Prometheus-Instances einrichten, die dieselben Metriken überwachen, und diese aus Gründen der Hochverfügbarkeit an einen einzigen Workspace in Amazon Managed Service für Prometheus senden, müssen Sie die Deduplizierung einrichten. Wenn Sie die Schritte zur Einrichtung der Deduplizierung nicht befolgen, werden Ihnen alle an Amazon Managed Service für Prometheus gesendeten Datenbeispiele, einschließlich doppelter Beispiele, in Rechnung gestellt. Anweisungen zur Einrichtung der Deduplizierung finden Sie unter Deduplizierung von Hochverfügbarkeitsmetriken, die an Amazon Managed Service für Prometheus gesendet werden.

Themen

- Richten Sie die Erfassung von einem neuen Prometheus-Server mithilfe des Befehls „Helm“ ein
- Richten Sie die Erfassung von einem vorhandenen Prometheus-Server in Kubernetes auf EC2 ein
- Richten Sie die Erfassung von einem vorhandenen Prometheus-Server in Kubernetes auf Fargate ein

Richten Sie die Erfassung von einem neuen Prometheus-Server mithilfe des Befehls „Helm“ ein

Mit den Anweisungen in diesem Abschnitt können Sie Amazon Managed Service für Prometheus schnell einrichten. Sie richten einen neuen Prometheus-Server in einem Amazon-EKS-Cluster ein und der neue Server verwendet eine Standardkonfiguration, um Metriken an Amazon Managed Service für Prometheus zu senden. Für diese Methode müssen die folgenden Voraussetzungen erfüllt sein:

- Sie müssen über einen Amazon EKS-Cluster verfügen, von dem aus der neue Prometheus-Server Metriken erfasst.
- Auf Ihrem Amazon EKS-Cluster muss ein Amazon EBS-CSI-Treiber installiert sein (von Helm erforderlich).
- Sie müssen Helm CLI 3.0 oder höher verwenden.

- Sie müssen einen Linux- oder macOS-Computer verwenden, um die Schritte in den folgenden Abschnitten auszuführen.

Schritt 1: Hinzufügen neuer Helm-Chart-Repositorys

Geben Sie die folgenden Befehle ein, um neue Helm-Chart-Repositorys hinzuzufügen. Weitere Informationen zu diesen Befehlen finden Sie unter 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
```

Schritt 2: Erstellen eines Prometheus-Namespace

Geben Sie den folgenden Befehl ein, um einen Prometheus-Namespace für den Prometheus-Server und andere Monitoring-Komponenten zu erstellen. *prometheus-namespace*Ersetzen Sie durch den Namen, den Sie für diesen Namespace haben möchten.

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

Schritt 3: Einrichten von IAM-Rollen für Servicekonten

Für die Onboarding-Methode, die wir dokumentieren, müssen Sie IAM-Rollen für Servicekonten im Amazon-EKS-Cluster verwenden, in dem der Prometheus-Server läuft.

Mit IAM-Rollen für Servicekonten können Sie eine IAM-Rolle mit einem Kubernetes-Servicekonto verknüpfen. Dieses Servicekonto kann dann AWS -Berechtigungen für die Container in einem beliebigen Pod bereitstellen, der dieses Servicekonto verwendet. Weitere Informationen finden Sie unter IAM-Rollen für Servicekonten.

Wenn Sie diese Rollen noch nicht eingerichtet haben, folgen Sie den Anweisungen unter Einrichten von Servicerollen für die Erfassung von Metriken aus Amazon-EKS-Clustern, um die Rollen einzurichten. Die Anweisungen in diesem Abschnitt erfordern die Verwendung von `eksctl`. Weitere Informationen finden Sie unter Erste Schritte mit Amazon Elastic Kubernetes Service – `eksctl`.

> **Note**
>
> Wenn Sie nicht auf EKS sind oder nur den Zugriffsschlüssel AWS und den geheimen Schlüssel verwenden, um auf Amazon Managed Service for Prometheus zuzugreifen, können Sie das `EKS-IAM-ROLE` basierte SigV4 nicht verwenden.

Schritt 4: Einrichten des neuen Servers und Beginn der Erfassung von Metriken

Gehen Sie wie folgt vor, um den neuen Prometheus-Server zu installieren, der Metriken an Ihren Workspace in Amazon Managed Service für Prometheus sendet.

Um einen neuen Prometheus-Server zu installieren, um Metriken an Ihren Workspace in Amazon Managed Service für Prometheus zu senden

1. Erstellen Sie mithilfe eines Texteditors eine Datei mit dem Namen `my_prometheus_values_yaml` mit folgenden Inhalten.

   - Ersetzen Sie *`IAM_PROXY_PROMETHEUS_ROLE_ARN`* durch den ARN von amp-iamproxy-ingest-role, die Sie in Einrichten von Servicerollen für die Erfassung von Metriken aus Amazon-EKS-Clustern erstellt haben.
   - *`WORKSPACE_ID`*Ersetzen Sie es durch die ID Ihres Amazon Managed Service for Prometheus Workspace.
   - *`REGION`*Ersetzen Sie es durch die Region Ihres Amazon Managed Service for Prometheus-Workspace.

```
## 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. Geben Sie den folgenden Befehl ein, um den Prometheus-Server zu starten.

   - Ersetzen Sie *prometheus-chart-name* durch Ihren Prometheus-Versionsnamen.
   - Ersetzen Sie es *prometheus-namespace* durch den Namen Ihres Prometheus-Namespaces.

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

> **Note**
>
> Sie können den `helm install`-Befehl auf viele Arten anpassen. Weitere Informationen finden Sie unter Helm-Installation in der Helm-Dokumentation.

Richten Sie die Erfassung von einem vorhandenen Prometheus-Server in Kubernetes auf EC2 ein

Amazon Managed Service für Prometheus unterstützt die Erfassung von Metriken von Prometheus-Servern in Clustern, auf denen Amazon EKS ausgeführt wird, und in selbstverwalteten Kubernetes-Clustern, die auf Amazon EC2 ausgeführt werden. Die detaillierten Anweisungen in diesem Abschnitt beziehen sich auf einen Prometheus-Server in einem Amazon-EKS-Cluster. Die Schritte für einen selbstverwalteten Kubernetes-Cluster auf Amazon EC2 sind dieselben, mit der Ausnahme, dass Sie die OIDC-Anbieter- und IAM-Rollen für Servicekonten selbst im Kubernetes-Cluster einrichten müssen.

In den Anweisungen in diesem Abschnitt wird Helm als Kubernetes-Paketmanager verwendet.

Themen

- Schritt 1: Einrichten von IAM-Rollen für Servicekonten
- Schritt 2: Aktualisieren Sie Ihren vorhandenen Prometheus-Server mithilfe von Helm

Schritt 1: Einrichten von IAM-Rollen für Servicekonten

Für die Onboarding-Methode, die wir dokumentieren, müssen Sie IAM-Rollen für Servicekonten im Amazon-EKS-Cluster verwenden, in dem der Prometheus-Server läuft. Diese Rollen werden als Servicerollen bezeichnet.

Mit Servicerollen können Sie eine IAM-Rolle mit einem Kubernetes-Servicekonto verknüpfen. Dieses Dienstkonto kann dann AWS Berechtigungen für die Container in jedem Pod gewähren, der dieses Dienstkonto verwendet. Weitere Informationen finden Sie unter IAM-Rollen für Servicekonten.

Wenn Sie diese Rollen noch nicht eingerichtet haben, folgen Sie den Anweisungen unter Einrichten von Servicerollen für die Erfassung von Metriken aus Amazon-EKS-Clustern, um die Rollen einzurichten.

Schritt 2: Aktualisieren Sie Ihren vorhandenen Prometheus-Server mithilfe von Helm

Die Anweisungen in diesem Abschnitt beinhalten die Einrichtung von Remote-Write und SigV4 zur Authentifizierung und Autorisierung des Prometheus-Servers für Remote–Write in Ihrem Workspace in Amazon Managed Service für Prometheus.

Verwenden von Prometheus Version 2.26.0 oder höher

Gehen Sie wie folgt vor, wenn Sie einen Helm-Chart mit einem Prometheus Server-Image der Version 2.26.0 oder höher verwenden.

So richten Sie Remote-Write von einem Prometheus-Server aus mithilfe von Helm-Chart ein

1. Erstellen Sie einen neuen Remote-Write-Abschnitt in Ihrer Helm-Konfigurationsdatei:

   - Ersetzen Sie `${IAM_PROXY_PROMETHEUS_ROLE_ARN}` durch den ARN von amp-iamproxy-ingest-role, die Sie in Schritt 1: Einrichten von IAM-Rollen für Servicekonten erstellt haben. Der Rollen-ARN sollte das Format `arn:aws:iam::`*`your account ID`*`:role/amp-iamproxy-ingest-role` haben.
   - Ersetzen Sie `${WORKSPACE_ID}` durch Ihre Workspace-ID von Amazon Managed Service für Prometheus.
   - Ersetzen Sie `${REGION}` durch die Region des Workspace in Amazon Managed Service für Prometheus (z. B. `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. Aktualisieren Sie Ihre bestehende Prometheus-Server-Konfiguration mit Helm:

   - Ersetzen Sie `prometheus-chart-name` durch Ihren Prometheus-Versionsnamen.
   - Ersetzen Sie `prometheus-namespace` durch den Kubernetes-Namespace, in dem Ihr Prometheus-Server installiert ist.
   - Ersetzen Sie `my_prometheus_values_yaml` durch den Pfad zu Ihrer Helm-Konfigurationsdatei.
   - Ersetzen Sie `current_helm_chart_version` durch die aktuelle Version Ihres Prometheus-Server-Helm-Charts. Sie können die aktuelle Chart-Version finden, indem Sie den Befehl Helmliste verwenden.

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

Verwenden früherer Versionen von Prometheus

Gehen Sie wie folgt vor, wenn Sie eine ältere Version von Prometheus als 2.26.0 verwenden. Bei diesen Schritten wird ein Sidecar-Ansatz verwendet, da frühere Versionen von Prometheus den AWS Signature Version 4-Signaturprozess (SigV4) nicht nativ unterstützen.AWS

Bei diesen Anleitungen wird davon ausgegangen, dass Sie Helm verwenden, um Prometheus bereitzustellen.

So richten Sie Remote-Write von einem Prometheus-Server aus ein

1. Erstellen Sie auf Ihrem Prometheus-Server eine neue Remote-Write-Konfiguration. Erstellen Sie zunächst eine neue Aktualisierungsdatei. Wir werden die Datei `amp_ingest_override_values.yaml` aufrufen.

   Fügen Sie der YAML-Datei die folgenden Werte hinzu.

```
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
```

Ersetzen Sie `${REGION}` durch die Region des Workspace in Amazon Managed Service für Prometheus.

Ersetzen Sie `${SERVICE_ACCOUNT_IAM_INGEST_ROLE_ARN}` durch den ARN von amp-iamproxy-ingest-role, die Sie in Schritt 1: Einrichten von IAM-Rollen für Servicekonten erstellt haben. Der Rollen-ARN sollte das Format `arn:aws:iam::`*`your account ID`*`:role/amp-iamproxy-ingest-role` haben.

Ersetzen Sie `${WORKSPACE_ID}` durch Ihre Workspace-ID.

2. Aktualisieren Sie Ihr Prometheus Helm-Chart. Suchen Sie zunächst nach dem Namen Ihres Helm-Charts, indem Sie den folgenden Befehl eingeben. Suchen Sie in der Ausgabe dieses Befehls nach einem Chart mit einem Namen, der `prometheus` enthält:

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

Geben Sie dann den folgenden Befehl ein.

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

*`prometheus-helm-chart-name`*Ersetzen Sie es durch den Namen der Prometheus-Helmkarte, die im vorherigen Befehl zurückgegeben wurde. Ersetzen Sie *`prometheus-namespace`* durch den Namen Ihres Namespaces.

Helm-Charts werden heruntergeladen

Wenn Sie Helm-Charts noch nicht lokal heruntergeladen haben, können Sie sie mit dem folgenden Befehl herunterladen.

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

Richten Sie die Erfassung von einem vorhandenen Prometheus-Server in Kubernetes auf Fargate ein

Amazon Managed Service für Prometheus unterstützt die Erfassung von Metriken von Prometheus-Servern in selbstverwalteten Kubernetes-Clustern, die auf Fargate ausgeführt werden. Um Metriken von Prometheus-Servern in Amazon-EKS-Clustern zu erfassen, die auf Fargate laufen, überschreiben Sie die Standardkonfigurationen in einer Konfigurationsdatei mit dem Namen amp_ingest_override_values.yaml wie folgt:

```
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
```

Installieren Sie Prometheus mithilfe der Overrides mit dem folgenden Befehl:

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

Beachten Sie, dass wir in der Helm-Chart-Konfiguration den Node Exporter und den Alert Manager deaktiviert und die Prometheus-Serverbereitstellung ausgeführt haben.

Sie können die Installation mit der folgenden Beispiel-Testabfrage überprüfen.

```
$ 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
```

Richten Sie Amazon Managed Service für Prometheus für Daten mit hoher Verfügbarkeit ein

Wenn Sie Daten an Amazon Managed Service für Prometheus senden, werden sie automatisch in AWS Availability Zones in der Region repliziert und Ihnen von einem Host-Cluster bereitgestellt, der Skalierbarkeit, Verfügbarkeit und Sicherheit bietet. Je nach Ihrer speziellen Konfiguration möchten Sie möglicherweise zusätzliche Ausfallsicherungen für hohe Verfügbarkeit hinzufügen. Es gibt zwei gängige Methoden, wie Sie Ihrem Setup zusätzliche Sicherheitsfunktionen für hohe Verfügbarkeit hinzufügen können:

- Wenn Sie mehrere Container oder Instances mit denselben Daten haben, können Sie diese Daten an Amazon Managed Service für Prometheus senden und die Daten automatisch deduplizieren lassen. Dadurch wird sichergestellt, dass Ihre Daten an Ihren Workspace in Amazon Managed Service für Prometheus gesendet werden.

  Weitere Informationen zur Deduplizierung von Hochverfügbarkeitsdaten finden Sie unter Deduplizierung von Hochverfügbarkeitsmetriken, die an Amazon Managed Service für Prometheus gesendet werden.
- Wenn Sie sicherstellen möchten, dass Sie Zugriff auf Ihre Daten haben, auch wenn die Region AWS nicht verfügbar ist, können Sie Ihre Metriken an einen zweiten Workspace in einer anderen Region senden.

  Weitere Informationen zum Senden von Metrikdaten an mehrere Workspaces finden Sie unter Verwenden Sie regionsübergreifende Arbeitsbereiche, um eine hohe Verfügbarkeit in Amazon Managed Service für Prometheus zu gewährleisten.

Themen

- Deduplizierung von Hochverfügbarkeitsmetriken, die an Amazon Managed Service für Prometheus gesendet werden
- Senden Sie mit Prometheus Hochverfügbarkeitsdaten an Amazon Managed Service für Prometheus
- Richten Sie mithilfe der Prometheus Operator Helm-Tabelle Hochverfügbarkeitsdaten für Amazon Managed Service for Prometheus ein
- Senden Sie Hochverfügbarkeitsdaten an Amazon Managed Service for Prometheus mit AWS Distro für OpenTelemetry
- Mit dem Prometheus-Community-Helm-Chart Hochverfügbarkeitsdaten an Amazon Managed Service für Prometheus senden
- Antworten auf häufig gestellte Fragen zur Hochverfügbarkeitskonfiguration in Amazon Managed Service für Prometheus
- Verwenden Sie regionsübergreifende Arbeitsbereiche, um eine hohe Verfügbarkeit in Amazon Managed Service für Prometheus zu gewährleisten

Deduplizierung von Hochverfügbarkeitsmetriken, die an Amazon Managed Service für Prometheus gesendet werden

Sie können Daten von mehreren Prometheus-Agenten (Prometheus-Instances, die im Agentenmodus ausgeführt werden) an Ihren Workspace in Amazon Managed Service für Prometheus senden. Wenn einige dieser Instances dieselben Metriken aufzeichnen und senden, haben Ihre Daten eine höhere Verfügbarkeit (auch wenn einer der Agenten keine Daten mehr sendet, empfängt der Workspace in Amazon Managed Service für Prometheus die Daten immer noch von einer anderen Instance). Sie möchten jedoch, dass Ihr Workspace in Amazon Managed Service für Prometheus die Metriken automatisch dedupliziert, sodass Sie die Metriken nicht mehrfach sehen und Ihnen nicht die mehrfache Datenerfassung und Speicherung in Rechnung gestellt werden.

Damit Amazon Managed Service für Prometheus automatisch Daten von mehreren Prometheus-Agenten deduplizieren kann, geben Sie der Gruppe von Agenten, die die doppelten Daten senden, einen einzelnen Clusternamen und jeder der Instances einen Replikatnamen. Der Clustername identifiziert, dass die Instances gemeinsam genutzte Daten haben, und der Replikatname ermöglicht es Amazon Managed Service für Prometheus, die Quelle jeder Metrik zu identifizieren. Die endgültigen gespeicherten Metriken beinhalten das Cluster-Etikett, nicht aber das Replikat. Die Metriken scheinen also aus einer einzigen Quelle zu stammen.

> **Note**
>
> Bestimmte Versionen von Kubernetes (1.28 und 1.29) geben möglicherweise ihre eigene Metrik mit einer Bezeichnung aus. `cluster` Dies kann zu Problemen mit Amazon Managed Service for Prometheus-Deduplizierung führen. Weitere Informationen finden Sie in den häufig gestellten Fragen zur Hochverfügbarkeit.

Die folgenden Themen zeigen, wie Daten gesendet werden und fügen die `__replica__` Labels `cluster` und hinzu, sodass Amazon Managed Service for Prometheus die Daten automatisch dedupliziert.

> **Important**
>
> Wenn Sie keine Deduplizierung einrichten, werden Ihnen alle Datenbeispiele in Rechnung gestellt, die an Amazon Managed Service für Prometheus gesendet werden. Diese Datenbeispiele enthalten doppelte Beispiele.

Senden Sie mit Prometheus Hochverfügbarkeitsdaten an Amazon Managed Service für Prometheus

Um eine Hochverfügbarkeitskonfiguration mit Prometheus einzurichten, müssen Sie externe Etiketten auf allen Instances einer Hochverfügbarkeitsgruppe anbringen, damit Amazon Managed Service für Prometheus sie identifizieren kann. Verwenden Sie das `cluster` Etikett, um einen Prometheus-Instance-Agenten als Teil einer Hochverfügbarkeitsgruppe zu identifizieren. Verwenden Sie das `__replica__` Etikett, um jedes Replikat in der Gruppe separat zu identifizieren. Sie müssen `__replica__` und `cluster` Etiketten anbringen, damit die Deduplizierung funktioniert.

> **Note**
>
> Das `__replica__` Etikett ist mit zwei Unterstrichen vor und nach dem Wort `replica` formatiert.

Beispiel: Codeausschnitte

In den folgenden Codeausschnitten identifiziert das `cluster`-Etiketten den Prometheus-Instance-Agenten `prom-team1` und das `_replica_`-Etikett identifiziert die Replikate `replica1` und `replica2`.

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

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

Da Amazon Managed Service für Prometheus Datenbeispiele von Hochverfügbarkeitsreplikaten mit diesen Etiketten speichert, entfernt es das `replica`-Etikett, wenn die Beispiele akzeptiert werden. Das bedeutet, dass Sie für Ihre aktuelle Serie nur ein 1:1 -Serien-Mapping haben werden und nicht eine Serie pro Replikat. Das `cluster` Etikett wird beibehalten.

> **Note**
>
> Bestimmte Versionen von Kubernetes (1.28 und 1.29) geben möglicherweise ihre eigene Metrik mit einer Bezeichnung aus. `cluster` Dies kann zu Problemen mit Amazon Managed Service for Prometheus-Deduplizierung führen. Weitere Informationen finden Sie in den häufig gestellten Fragen zur Hochverfügbarkeit.

Richten Sie mithilfe der Prometheus Operator Helm-Tabelle Hochverfügbarkeitsdaten für Amazon Managed Service for Prometheus ein

Um eine Hochverfügbarkeitskonfiguration mit dem Prometheus-Operator in Helm einzurichten, müssen Sie alle Instances einer Hochverfügbarkeitsgruppe mit externen Labels versehen, damit Amazon Managed Service for Prometheus sie identifizieren kann. Sie müssen auch die Attribute `replicaExternalLabelName` und `externalLabels` im Prometheus-Operator-Helm-Chart festlegen.

Beispiel: YAML-Header

Im folgenden YAML-Header `cluster` wird `externalLabel` hinzugefügt, um einen Prometheus-Instance-Agenten als Teil einer Hochverfügbarkeitsgruppe zu identifizieren, und `replicaExternalLabels` identifiziert jedes Replikat in der Gruppe.

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

> **Note**
>
> Bestimmte Versionen von Kubernetes (1.28 und 1.29) geben möglicherweise ihre eigene Metrik mit einer Bezeichnung aus. `cluster` Dies kann zu Problemen mit Amazon Managed Service for Prometheus-Deduplizierung führen. Weitere Informationen finden Sie in den häufig gestellten Fragen zur Hochverfügbarkeit.

Senden Sie Hochverfügbarkeitsdaten an Amazon Managed Service for Prometheus mit AWS Distro für OpenTelemetry

AWS Distro for OpenTelemetry (ADOT) ist eine sichere und produktionsbereite Distribution des Projekts. OpenTelemetry ADOT stellt Ihnen Quell-APIs, Bibliotheken und Agenten zur Verfügung, sodass Sie verteilte Traces und Metriken für die Anwendungsüberwachung sammeln können. Informationen zu ADOT finden Sie unter Über AWS Distro for Open Telemetry.

Um ADOT mit einer Hochverfügbarkeitskonfiguration einzurichten, müssen Sie ein ADOT-Collector-Container-Image konfigurieren und die externen Labels auf den Prometheus Remote Write `cluster` `__replica__` Exporter anwenden. AWS Dieser Exporter sendet Ihre gesammelten Metriken über den `remote_write`-Endpunkt an Ihren Workspace in Amazon Managed Service für Prometheus. Wenn Sie diese Etiketten auf dem Remote Write-Exporter einrichten, verhindern Sie, dass doppelte Metriken gespeichert werden, während redundante Replikate ausgeführt werden. Weitere Informationen zum AWS Prometheus Remote Write Exporter finden Sie unter Erste Schritte mit Prometheus Remote Write Exporter für Amazon Managed Service for Prometheus.

> **Note**
>
> Bestimmte Versionen von Kubernetes (1.28 und 1.29) geben möglicherweise ihre eigene Metrik mit einer Bezeichnung aus. `cluster` Dies kann zu Problemen mit der Deduplizierung durch Amazon Managed Service for Prometheus führen. Weitere Informationen finden Sie in den häufig gestellten Fragen zur Hochverfügbarkeit.

Mit dem Prometheus-Community-Helm-Chart Hochverfügbarkeitsdaten an Amazon Managed Service für Prometheus senden

Um eine Hochverfügbarkeitskonfiguration mit dem Prometheus-Community-Helm-Chart einzurichten, müssen Sie externe Etiketten auf allen Instances einer Hochverfügbarkeitsgruppe anbringen, damit Amazon Managed Service für Prometheus sie identifizieren kann. Hier ist ein Beispiel dafür, wie Sie `external_labels` zur einer einzelnen Instance von Prometheus aus dem Helm-Chart aus der Prometheus-Community hinzufügen könnten.

```
server:
global:
  external_labels:
      cluster: monitoring-cluster
      __replica__: replica-1
```

> **Note**
>
> Wenn Sie mehrere Replikate benötigen, müssen Sie das Diagramm mehrmals mit unterschiedlichen Replikatwerten bereitstellen, da Sie mit dem Helm-Chart der Prometheus-Community den Replikatwert nicht dynamisch festlegen können, wenn Sie die Anzahl der Replikate direkt von der Controller-Gruppe aus erhöhen. Wenn Sie es vorziehen, dass das `replica` Etikett automatisch festgelegt wird, verwenden Sie das Helm-Chart des Prometheus-Operators.

> **Note**
>
> Bestimmte Versionen von Kubernetes (1.28 und 1.29) geben möglicherweise ihre eigene Metrik mit einer Bezeichnung aus. `cluster` Dies kann zu Problemen mit der Deduplizierung durch Amazon Managed Service for Prometheus führen. Weitere Informationen finden Sie in den häufig gestellten Fragen zur Hochverfügbarkeit.

Antworten auf häufig gestellte Fragen zur Hochverfügbarkeitskonfiguration in Amazon Managed Service für Prometheus

Sollte ich den Wert angeben __replica__ in ein anderes Etikett, um die Probenpunkte zu verfolgen?

In einer Hochverfügbarkeitsumgebung stellt Amazon Managed Service für Prometheus sicher, dass Datenbeispiele nicht dupliziert werden, indem ein führender Anbieter im Cluster von Prometheus-Instances ausgewählt wird. Wenn das Leader-Replikat 30 Sekunden lang keine Datenbeispiele mehr sendet, macht Amazon Managed Service für Prometheus automatisch eine andere Prometheus-Instance zu einem Leader-Replikat und erfasst Daten vom neuen Leader-Replikat, einschließlich aller fehlenden Daten. Daher lautet die Antwort nein, es wird nicht empfohlen. Dies kann zu folgenden Problemen führen:

- Die Abfrage von `count` in PromQL kann während der Zeit der Wahl eines neuen Leaders zu einem höheren Wert als erwartet führen.
- Die Anzahl von `active series` wird während der Wahl eines neuen Leaders erhöht und erreicht `active series limits`. Weitere Informationen finden Sie unter AMP-Quoten.

Kubernetes scheint sein eigenes zu haben Cluster Label und dedupliziert meine Metriken nicht. Wie lässt sich dieses Problem beheben?

In Kubernetes 1.28 `apiserver_storage_size_bytes` wurde eine neue Metrik mit einer Bezeichnung eingeführt. `cluster` Dies kann zu Problemen mit der Deduplizierung in Amazon Managed Service for Prometheus führen, was vom Label abhängt. `cluster` In Kubernetes 1.3 wurde das Label umbenannt in `storage-cluster_id` (es wird auch in späteren Patches 1.28 und 1.29 umbenannt). Wenn Ihr Cluster diese Metrik mit dem `cluster` Label ausgibt, kann Amazon Managed Service for Prometheus die zugehörige Zeitreihe nicht deduplizieren. Wir empfehlen Ihnen, Ihren Kubernetes-Cluster auf die neueste gepatchte Version zu aktualisieren, um dieses Problem zu vermeiden. Alternativ können Sie das `cluster` Label auf Ihrer `apiserver_storage_size_bytes` Metrik umbenennen, bevor Sie sie in Amazon Managed Service for Prometheus aufnehmen.

> **Note**
>
> Weitere Informationen zur Umstellung auf Kubernetes finden Sie unter Umbenennen des Label-Clusters in storage_cluster_id für die Metrik apiserver_storage_size_bytes im Kubernetes-Projekt. GitHub

Verwenden Sie regionsübergreifende Arbeitsbereiche, um eine hohe Verfügbarkeit in Amazon Managed Service für Prometheus zu gewährleisten

Um deinen Daten eine regionsübergreifende Verfügbarkeit zu verleihen, kannst du Metriken an mehrere Workspaces in AWS verschiedenen Regionen senden. Prometheus unterstützt sowohl das Schreiben mehrerer Autoren als auch regionsübergreifendes Schreiben.

Das folgende Beispiel zeigt, wie Sie einen Prometheus-Server einrichten, der im Agent-Modus läuft, um mit Helm Metriken an zwei Workspaces in verschiedenen Regionen zu senden.

```
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]
```

Abfragen Ihrer Prometheus-Metriken

Jetzt, da die Metriken in den Workspace erfasst wurden, können Sie sie abfragen.

Um Dashboards mit visuellen Darstellungen Ihrer Kennzahlen zu erstellen, können Sie einen Service wie Amazon Managed Grafana verwenden. Amazon Managed Grafana (oder eine eigenständige Instanz von Grafana) kann eine grafische Oberfläche erstellen, die Ihre Metriken in einer Vielzahl von Darstellungsarten anzeigt. Weitere Informationen zu Amazon Managed Grafana finden Sie im Amazon Managed Grafana-Benutzerhandbuch.

Sie können auch einmalige Abfragen erstellen, Ihre Daten untersuchen oder Ihre eigenen Anwendungen schreiben, die Ihre Metriken mithilfe direkter Abfragen verwenden. Direkte Abfragen verwenden die Amazon Managed Service for Prometheus API und die standardmäßige Prometheus-Abfragesprache PromQL, um Daten aus Ihrem Prometheus-Workspace abzurufen. Weitere Informationen zu PromQL und seiner Syntax finden Sie unter Abfragen von Prometheus in der Prometheus-Dokumentation.

Themen

- PromQL-Spickzettel
- Grundlegende Selektoren
- Selektoren für Bereichsvektoren
- Aggregationsoperatoren
- Allgemeine Funktionen
- Binäre Operatoren
- Praktische Beispiele für Abfragen
- Schützen Sie Ihre metrischen Abfragen
- Einrichten von Amazon Managed Grafana für die Verwendung mit Amazon Managed Service für Prometheus
- Richten Sie Grafana Open Source oder Grafana Enterprise für die Verwendung mit Amazon Managed Service für Prometheus ein
- Abfrage mithilfe von Grafana, die in einem Amazon-EKS-Cluster ausgeführt wird
- Abfragen mithilfe von APIs Prometheus-compatible
- Rufen Sie für jede Abfrage Statistiken über Ihre Abfragenutzung ab

PromQL-Spickzettel

Verwenden Sie diesen PromQL-Spickzettel (Prometheus Query Language) als Kurzreferenz, wenn Sie Kennzahlen in Ihrem Amazon Managed Service for Prometheus-Workspace abfragen. Mit PromQL können Sie mithilfe der funktionalen Abfragesprache Zeitreihendaten in Echtzeit auswählen und aggregieren.

Weitere Informationen zu PromQL finden Sie im PromQL Cheat Sheet auf der Website. PromLabs

Grundlegende Selektoren

Wählen Sie Zeitreihen nach Metriknamen und Label-Matchern aus:

```
# 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
```

Selektoren für Bereichsvektoren

Wählen Sie im Laufe der Zeit einen Bereich von Samples aus:

```
# 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]
```

Aggregationsoperatoren

Aggregieren Sie Daten über mehrere Zeitreihen hinweg:

```
# 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
```

Allgemeine Funktionen

Wenden Sie Funktionen an, um Ihre Daten zu transformieren:

```
# 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)
```

Binäre Operatoren

Führen Sie arithmetische und logische Operationen durch:

```
# 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)
```

Praktische Beispiele für Abfragen

Allgemeine Überwachungsanfragen, die Sie in Ihrem Amazon Managed Service for Prometheus Workspace verwenden können:

```
# 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))
```

Schützen Sie Ihre metrischen Abfragen

Amazon Managed Service für Prometheus bietet Möglichkeiten, Sie bei der sicheren Abfrage Ihrer Messwerte zu unterstützen.

Verwenden AWS PrivateLink mit Amazon Managed Service für Prometheus

Der Netzwerkverkehr für die Abfrage von Metriken in Amazon Managed Service for Prometheus kann über einen öffentlichen Internet-Endpunkt oder über einen VPC-Endpunkt erfolgen. AWS PrivateLink Bei der Verwendung AWS PrivateLink wird der Netzwerkverkehr von Ihren VPCs innerhalb des AWS Netzwerks gesichert, ohne dass er über das öffentliche Internet übertragen wird. Informationen zum Erstellen eines AWS PrivateLink VPC-Endpunkts für Amazon Managed Service for Prometheus finden Sie unter. Verwendung von Amazon Managed Service für Prometheus mit Schnittstellen-VPC-Endpunkten

Authentifizierung und Autorisierung

AWS Identity and Access Management ist ein Webservice, mit dem Sie den Zugriff auf Ressourcen sicher kontrollieren können. AWS Sie verwenden IAM, um zu steuern, wer authentifiziert (angemeldet) und autorisiert (Berechtigungen besitzt) ist, Ressourcen zu nutzen. Amazon Managed Service für Prometheus ist in IAM integriert, um Ihnen zu helfen, Ihre Daten zu schützen. Wenn Sie Amazon Managed Service für Prometheus einrichten, müssen Sie einige IAM-Rollen erstellen, die es Grafana-Servern ermöglichen, Metriken abzufragen, die in Workspaces von Amazon Managed Service für Prometheus gespeichert sind. Weitere Informationen zu IAM finden Sie unter Was ist IAM?.

Eine weitere AWS Sicherheitsfunktion, die Ihnen bei der Einrichtung von Amazon Managed Service für Prometheus helfen kann, ist der AWS Signature Version 4-Signaturprozess (AWS Sigv4). Signature Version 4 ist der Prozess zum Hinzufügen von Authentifizierungsinformationen zu AWS Anfragen, die über HTTP gesendet werden. Aus Sicherheitsgründen AWS müssen die meisten Anfragen mit einem Zugriffsschlüssel signiert werden, der aus einer Zugriffsschlüssel-ID

und einem geheimen Zugriffsschlüssel besteht. Diese beiden Schlüssel werden in der Regel als Sicherheitsanmeldeinformationen bezeichnet. Weitere Informationen über SigV4 finden Sie unter Signatur Version 4-Signierungsprozess.

Einrichten von Amazon Managed Grafana für die Verwendung mit Amazon Managed Service für Prometheus

Amazon Managed Grafana ist ein vollständig verwalteter Service für Open-Source-Grafana, der die Verbindung zu Open-Source-ISVs und AWS Services für die skalierbare Visualisierung und Analyse Ihrer Datenquellen vereinfacht.

Amazon Managed Service für Prometheus unterstützt die Verwendung von Amazon Managed Grafana zum Abrufen von Abfragemetriken in einem Workspace. In der Konsole von Amazon Managed Grafana können Sie einen Workspace in Amazon Managed Service für Prometheus als Datenquelle hinzufügen, indem Sie Ihre vorhandenen Konten für Amazon Managed Service für Prometheus ermitteln. Amazon Managed Grafana verwaltet die Konfiguration der Authentifizierungsdaten, die für den Zugriff auf Amazon Managed Service für Prometheus erforderlich sind. Detaillierte Anweisungen zum Herstellen einer Verbindung zu Amazon Managed Service für Prometheus von Amazon Managed Grafana aus finden Sie in den Anweisungen im Benutzerhandbuch von Amazon Managed Grafana.

Sie können Ihre Alarme für Amazon Managed Service für Prometheus auch in Amazon Managed Grafana anzeigen. Anweisungen zum Einrichten der Integration mit Alarmen finden Sie unter Integrieren Sie Benachrichtigungen in Amazon Managed Grafana oder Open-Source-Grafana.

Verbindung zu Amazon Managed Grafana in einer privaten VPC herstellen

Amazon Managed Service für Prometheus bietet einen Service-Endpunkt, zu dem Amazon Managed Grafana eine Verbindung herstellen kann, wenn Metriken und Alarme abgefragt werden.

Sie können Amazon Managed Grafana für die Verwendung einer privaten VPC konfigurieren (Einzelheiten zur Einrichtung einer privaten VPC in Grafana finden Sie unter Herstellen einer Verbindung zu Amazon VPC im Benutzerhandbuch von Amazon Managed Grafana). Abhängig von den Einstellungen hat diese VPC möglicherweise keinen Zugriff auf den Service-Endpunkt von Amazon Managed Service für Prometheus.

Um Amazon Managed Service für Prometheus als Datenquelle zu einem Workspace in Amazon Managed Grafana-hinzuzufügen, der für die Verwendung einer bestimmten privaten VPC konfiguriert

ist, müssen Sie zuerst Amazon Managed Service für Prometheus mit derselben VPC verbinden, indem Sie einen VPC-Endpunkt erstellen. Weitere Informationen zum Erstellen eines VPC-Endpunkts finden Sie unter Erstellen eines Schnittstellen-VPC-Endpunkts für Amazon Managed Service für Prometheus.

Richten Sie Grafana Open Source oder Grafana Enterprise für die Verwendung mit Amazon Managed Service für Prometheus ein

Sie können eine Instanz von Grafana verwenden, um Ihre Metriken in Amazon Managed Service for Prometheus abzufragen. In diesem Thema erfahren Sie, wie Sie mithilfe einer eigenständigen Instanz von Grafana Metriken von Amazon Managed Service for Prometheus abfragen.

Voraussetzungen

Grafana-Instanz — Sie benötigen eine Grafana-Instance, die sich bei Amazon Managed Service for Prometheus authentifizieren kann.

Amazon Managed Service für Prometheus unterstützt die Verwendung von Grafana-Version 7.3.5 und höher zur Abfrage von Metriken in einem Workspace. Die Versionen 7.3.5 und höher bieten Unterstützung für die Authentifizierung mit AWS Signature Version 4 (Sigv4).

Um Ihre Grafana-Version zu überprüfen, geben Sie den folgenden Befehl ein und *`grafana_install_directory`* ersetzen Sie ihn durch den Pfad zu Ihrer Grafana-Installation:

```
grafana_install_directory/bin/grafana-server -v
```

Wenn Sie noch kein eigenständiges Grafana haben oder eine neuere Version benötigen, können Sie eine neue Instanz installieren. Anweisungen zum Einrichten eines eigenständigen Grafana finden Sie unter Grafana installieren in der Grafana-Dokumentation. Informationen zu den ersten Schritten mit Grafana finden Sie unter Erste Schritte mit Grafana in der Grafana-Dokumentation.

AWS-Konto— Sie müssen über die richtigen Berechtigungen verfügen, um auf Ihre Amazon Managed Service for Prometheus-Metriken zugreifen zu können. AWS-Konto

Um Grafana für die Zusammenarbeit mit Amazon Managed Service for Prometheus einzurichten, müssen Sie bei einem Konto angemeldet sein, das über die AmazonPrometheusQueryAccessRichtlinie oder die`aps:QueryMetrics`, `aps:GetMetricMetadataaps:GetSeries`, und Berechtigungen verfügt. `aps:GetLabels` Weitere Informationen finden Sie unter IAM-Berechtigungen und Richtlinien.

Im nächsten Abschnitt wird die Einrichtung der Authentifizierung von Grafana aus ausführlicher beschrieben.

Schritt 1: Einrichten von AWS SigV4

Amazon Managed Service for Prometheus arbeitet mit AWS Identity and Access Management (IAM) zusammen, um alle Aufrufe von Prometheus-APIs mit IAM-Anmeldeinformationen zu sichern. Standardmäßig geht die Prometheus-Datenquelle in Grafana davon aus, dass Prometheus keine Authentifizierung erfordert. Damit Grafana die Authentifizierungs- und Autorisierungsfunktionen von Amazon Managed Service für Prometheus nutzen kann, müssen Sie die SigV4-Authentifizierungsunterstützung in der Grafana-Datenquelle aktivieren. Folgen Sie den Schritten auf dieser Seite, wenn Sie einen selbstverwalteten Grafana-Open-Source-Server oder einen Grafana-Unternehmensserver verwenden. Wenn Sie Amazon Managed Grafana verwenden, ist die SigV4-Authentifizierung vollständig automatisiert. Weitere Informationen zu Amazon Managed Grafana finden Sie unter Was ist Amazon Managed Grafana?

Um SigV4 auf Grafana zu aktivieren, starten Sie Grafana mit den `AWS_SDK_LOAD_CONFIG` und `GF_AUTH_SIGV4_AUTH_ENABLED` Umgebungsvariablen, die auf `true` eingestellt sind. Die `GF_AUTH_SIGV4_AUTH_ENABLED` Umgebungsvariable überschreibt die Standardkonfiguration für Grafana, um die SigV4-Unterstützung zu aktivieren. Weitere Informationen finden Sie unter Konfiguration in der Grafana-Dokumentation.

Linux

Um SigV4 auf einem eigenständigen Grafana unter Linux zu aktivieren, geben Sie die folgenden Befehle ein.

```
export AWS_SDK_LOAD_CONFIG=true
```

```
export GF_AUTH_SIGV4_AUTH_ENABLED=true
```

```
cd grafana_install_directory
```

```
./bin/grafana-server
```

Windows

Geben Sie die folgenden Befehle ein, um SigV4 auf einem eigenständigen Grafana unter Windows mithilfe der Windows-Eingabeaufforderung zu aktivieren.

```
set AWS_SDK_LOAD_CONFIG=true
```

```
set GF_AUTH_SIGV4_AUTH_ENABLED=true
```

```
cd grafana_install_directory
```

```
.\bin\grafana-server.exe
```

Schritt 2: Fügen Sie die Prometheus-Datenquelle in Grafana hinzu

In den folgenden Schritten wird erklärt, wie Sie die Prometheus-Datenquelle in Grafana einrichten, um Ihre Metriken von Amazon Managed Service für Prometheus abzufragen.

Um die Prometheus-Datenquelle zu Ihrem Grafana hinzuzufügen

1. Öffnen Sie die Grafana-Konsole.
2. Wählen Sie unter Konfigurationen die Option Datenquellen aus.
3. Wählen Sie Datenquelle hinzufügen aus.
4. Wählen Sie Prometheus aus.
5. Geben Sie für die HTTP-URL die Endpunkt – URL-Abfrage an, die auf der Workspace-Detailseite in der Konsole von Amazon Managed Service für Prometheus angezeigt wird.
6. Entfernen Sie in der HTTP-URL, die Sie gerade angegeben haben, die `/api/v1/query` Zeichenfolge, die an die URL angehängt ist, da die Prometheus-Datenquelle sie automatisch anhängt.

   Die richtige URL sollte ähnlich aussehen wie. https://aps-workspaces.us-west-2.amazonaws.com/workspaces/ws-1234a5b6-78cd-901e-2fgh-3i45j6k178l9
7. Wählen Sie unter Auth den Schalter für SigV4-Authentifizierung aus, um sie zu aktivieren.
8. Sie können die SigV4-Autorisierung entweder konfigurieren, indem Sie Ihre langfristigen Anmeldeinformationen direkt in Grafana angeben, oder indem Sie eine Standardanbieterkette verwenden. Wenn Sie Ihre langfristigen Anmeldeinformationen direkt angeben, können Sie schneller starten. In den folgenden Schritten finden Sie zunächst diese Anweisungen. Sobald Sie mit der Verwendung von Grafana mit Amazon Managed Service für Prometheus vertraut sind, empfehlen wir Ihnen, eine Standardanbieterkette zu verwenden, da diese

eine bessere Flexibilität und Sicherheit bietet. Weitere Informationen zum Einrichten Ihrer Standardanbieterkette finden Sie unter Angeben von Anmeldeinformationen.

- Gehen Sie wie folgt vor, um Ihre langfristigen Anmeldeinformationen direkt zu verwenden:

  a. Wählen Sie unter SigV4-Authentifizierungsdetails für Authentifizierungsanbieter die Option Zugangs- und geheimer Schlüssel aus.

  b. Was den Zugriffsschlüssel-ID angeht, geben Sie Ihre AWS Zugriffsschlüssel-ID ein.

  c. Was den geheimen Zugriffsschlüssel angeht, geben Sie Ihren geheimen AWS -Zugriffsschlüssel ein.

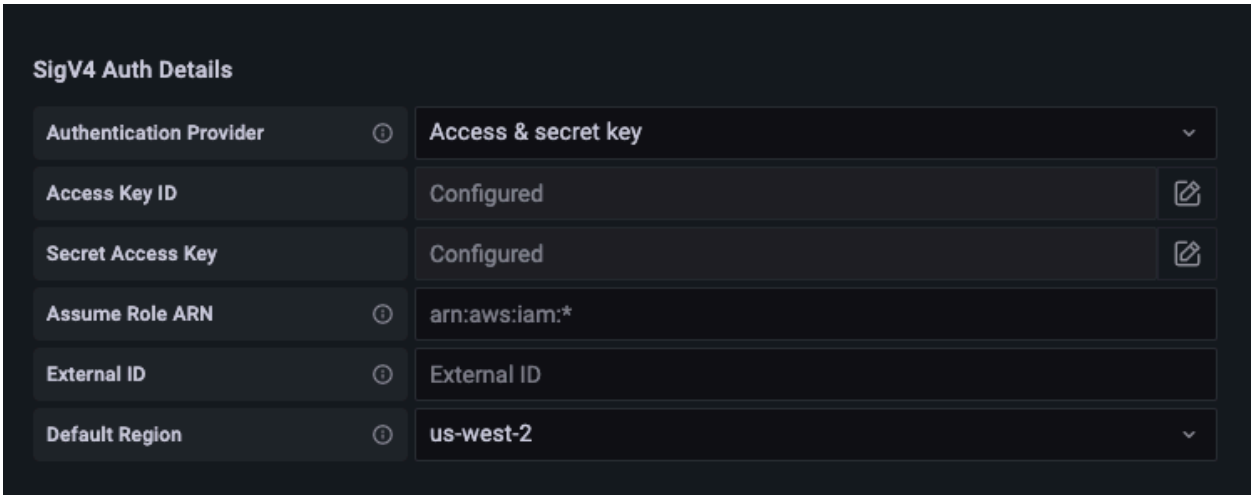
  d. Lassen Sie die Felder ARN-Rolle übernehmen und Externe ID leer.

  e. Wählen Sie unter Standardregion die Region Ihres Workspace in Amazon Managed Service für Prometheus aus. Diese Region sollte mit der Region übereinstimmen, die in der URL enthalten ist, die Sie in Schritt 5 angegeben haben.

  f. Wählen Sie Speichern & Testen aus.

  Sie sollten die folgende Nachricht sehen: Die Datenquelle funktioniert

  Der folgende Screenshot zeigt die Einstellung „Zugriffsschlüssel, geheimer Schlüssel, SigV4-Authentifizierungsdetails“.



- Gehen Sie wie folgt vor, um stattdessen eine Standardanbieterkette zu verwenden (empfohlen für eine Produktionsumgebung):

  a. Wählen Sie unter SigV4-Authentifizierungsdetails für Authentifizierungsanbieter die Option AWS -SDK-Standard aus.

  b. Lassen Sie die Felder ARN-Rolle übernehmen und Externe ID leer.

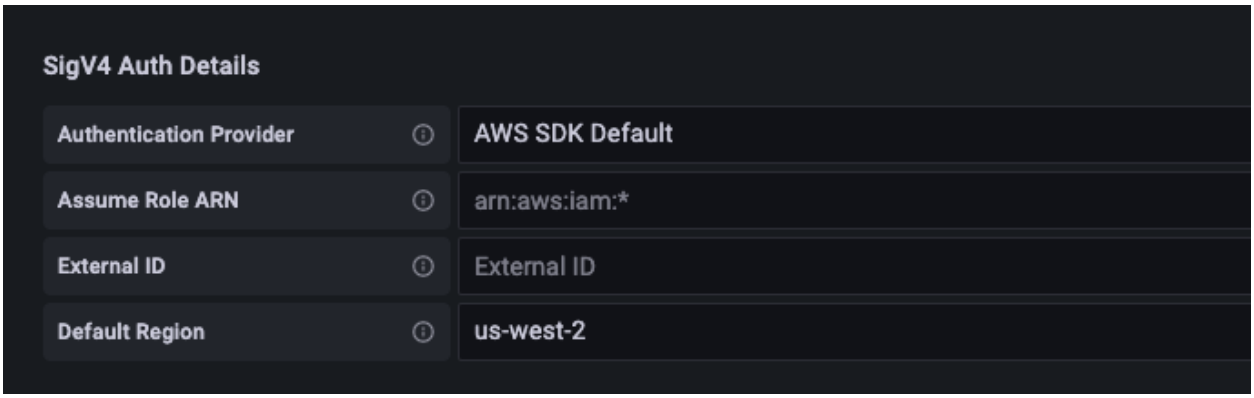
c. Wählen Sie unter Standardregion die Region Ihres Workspace in Amazon Managed Service für Prometheus aus. Diese Region sollte mit der Region übereinstimmen, die in der URL enthalten ist, die Sie in Schritt 5 angegeben haben.

d. Wählen Sie Speichern & Testen aus.

Sie sollten die folgende Nachricht sehen: Die Datenquelle funktioniert

Wenn Sie diese Meldung nicht sehen, finden Sie im nächsten Abschnitt Tipps zur Fehlerbehebung beim Herstellen einer Verbindung.

Der folgende Screenshot zeigt die SDK-Standardeinstellung für SigV4-Authentifizierungsdetails.



9. Testen Sie eine PromQL-Abfrage anhand der neuen Datenquelle:

a. Wählen Sie Explore.

b. Führen Sie eine PromQL-Beispielabfrage aus, wie zum Beispiel:

```
prometheus_tsdb_head_series
```

Schritt 3: (optional) Fehlerbehebung, wenn Save Test nicht funktioniert &

Wenn im vorherigen Verfahren bei der Auswahl von Speichern & Testen ein Fehler angezeigt wird, überprüfen Sie Folgendes.

HTTP-Fehler nicht gefunden

Vergewissern Sie sich, dass die Workspace-ID in der URL korrekt ist.

HTTP-Fehler verboten

Dieser Fehler bedeutet, dass die Anmeldeinformationen nicht gültig sind. Überprüfen Sie, ob Folgendes der Fall ist:

- Vergewissern Sie sich, dass die unter Standardregion angegebene Region korrekt ist.
- Überprüfen Sie Ihre Anmeldeinformationen auf Tippfehler.
- Stellen Sie sicher, dass die von Ihnen verwendeten Anmeldeinformationen die AmazonPrometheusQueryAccessRichtlinie enthalten. Weitere Informationen finden Sie unter IAM-Berechtigungen und Richtlinien.
- Stellen Sie sicher, dass die von Ihnen verwendeten Anmeldeinformationen Zugriff auf diesen Workspace in Amazon Managed Service für Prometheus haben.

HTTP-Fehler, schlechtes Gateway

Sehen Sie sich das Grafana-Serverprotokoll an, um diesen Fehler zu beheben. Weitere Informationen finden Sie unter Fehlerbehebung in der Microsoft-Dokumentation.

Wie Sie sehen**`Error http: proxy error: NoCredentialProviders: no valid providers in chain`**, konnte die standardmäßige Anbieterkette für Anmeldeinformationen keine gültigen AWS Anmeldeinformationen finden, die verwendet werden könnten. Stellen Sie sicher, dass Sie Ihre Anmeldeinformationen wie unter Anmeldeinformationen angeben beschrieben eingerichtet haben. Wenn Sie eine gemeinsam genutzte Konfiguration verwenden möchten, stellen Sie sicher, dass die `AWS_SDK_LOAD_CONFIG` Umgebung auf `true` eingestellt ist.

Abfrage mithilfe von Grafana, die in einem Amazon-EKS-Cluster ausgeführt wird

Amazon Managed Service für Prometheus unterstützt die Verwendung von Grafana-Version 7.3.5 und höher zur Abfrage von Metriken in einem Workspace in Amazon Managed Service für Prometheus. Die Versionen 7.3.5 und höher bieten Unterstützung für die Authentifizierung mit AWS Signature Version 4 (Sigv4).

Um Grafana für die Zusammenarbeit mit Amazon Managed Service for Prometheus einzurichten, müssen Sie bei einem Konto angemeldet sein, das über die AmazonPrometheusQueryAccessRichtlinie oder die`aps:QueryMetrics`,

`aps:GetMetricMetadataaps:GetSeries`, und Berechtigungen verfügt. `aps:GetLabels` Weitere Informationen finden Sie unter IAM-Berechtigungen und Richtlinien.

Einrichten AWS SigV4

Grafana hat eine neue Funktion zur Unterstützung der Authentifizierung mit AWS Signature Version 4 (Sigv4) hinzugefügt. Weitere Informationen finden Sie unter Signatur Version 4-Signierungsprozess. Diese Feature ist auf Grafana-Servern standardmäßig aktiviert. Die folgenden Anweisungen zur Aktivierung dieser Feature basieren darauf, dass Sie Helm verwenden, um Grafana auf einem Kubernetes-Cluster zu benutzen.

Um SigV4 auf Ihrem Server Grafana 7.3.5 oder höher zu aktivieren

1. Erstellen Sie eine neue Aktualisierungsdatei, um Ihre Grafana-Konfiguration zu überschreiben, und geben Sie ihr den Namen `amp_query_override_values.yaml`.
2. Kopieren Sie folgenden Inhalt in die Datei. Speichern Sie anschließend die Datei. *`account-id`*Ersetzen Sie durch die AWS Konto-ID, auf der der Grafana-Server läuft.

```
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
```

   In dem Inhalt dieser YAML-Datei ist `amp-iamproxy-query-role` der Name der Rolle, die Sie im nächsten Abschnitt, IAM-Rollen für Servicekonten einstellen erstellen werden. Sie können diese Rolle durch Ihren eigenen Rollennamen ersetzen, wenn Sie bereits eine Rolle für die Abfrage Ihres Workspace erstellt haben.

Sie werden diese Datei später verwenden, in Aktualisieren Sie den Grafana-Server mit Helm.

IAM-Rollen für Servicekonten einstellen

Wenn Sie einen Grafana-Server in einem Amazon-EKS-Cluster verwenden, empfehlen wir Ihnen, IAM-Rollen für Servicekonten, auch Servicerollen genannt, für Ihre Zugriffskontrolle zu verwenden. Wenn Sie dies tun, um eine IAM-Rolle einem Kubernetes-Dienstkonto zuzuordnen, kann das

Dienstkonto dann AWS Berechtigungen für die Container in jedem Pod gewähren, der dieses Dienstkonto verwendet. Weitere Informationen finden Sie unter IAM-Rollen für Servicekonten.

Wenn Sie diese Servicerollen noch nicht für Abfragen eingerichtet haben, folgen Sie den Anweisungen unter Richten Sie IAM-Rollen für Servicekonten zur Abfrage von Metriken ein, um die Rollen einzurichten.

Anschließend müssen Sie das Grafana-Servicekonto in den Bedingungen der Vertrauensbeziehung hinzufügen.

Um das Grafana-Servicekonto zu den Bedingungen der Vertrauensbeziehung hinzuzufügen

1. Bestimmen Sie in einem Terminalfenster den Namespace und den Namen des Servicekontos für Ihren Grafana-Server. Sie könnten z. B. den folgenden Befehl verwenden: .

   ```
   kubectl get serviceaccounts -n grafana_namespace
   ```

2. Öffnen Sie in der Amazon-EKS-Konsole die IAM-Rolle für Servicekonten, die dem EKS-Cluster zugeordnet ist.
3. Wählen Sie Vertrauensstellung bearbeiten aus.
4. Aktualisieren Sie die Bedingung, um den Grafana-Namespace und den Namen des Grafana-Servicekontos einzuschließen, die Sie in der Ausgabe des Befehls in Schritt 1 gefunden haben. Im Folgenden wird ein -Beispiel gezeigt.

   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. Wählen Sie Update trust policy (Vertrauensrichtlinie aktualisieren) aus.

Aktualisieren Sie den Grafana-Server mit Helm

In diesem Schritt wird der Grafana-Server aktualisiert, sodass er die Einträge verwendet, die Sie der `amp_query_override_values.yaml` Datei im vorherigen Abschnitt hinzugefügt haben.

Führen Sie die folgenden Befehle aus. Weitere Informationen zu Helm-Charts für Grafana finden Sie unter Helm-Charts für Grafana Community Kubernetes.

```
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
```

Fügen Sie die Prometheus-Datenquelle in Grafana hinzu

In den folgenden Schritten wird erklärt, wie Sie die Prometheus-Datenquelle in Grafana einrichten, um Ihre Metriken von Amazon Managed Service für Prometheus abzufragen.

Um die Prometheus-Datenquelle zu Ihrem Grafana hinzuzufügen

1. Öffnen Sie die Grafana-Konsole.
2. Wählen Sie unter Konfigurationen die Option Datenquellen aus.
3. Wählen Sie Datenquelle hinzufügen aus.
4. Wählen Sie Prometheus aus.

5. Geben Sie für die HTTP-URL die Endpunkt – URL-Abfrage an, die auf der Workspace-Detailseite in der Konsole von Amazon Managed Service für Prometheus angezeigt wird.
6. Entfernen Sie in der HTTP-URL, die Sie gerade angegeben haben, die `/api/v1/query` Zeichenfolge, die an die URL angehängt ist, da die Prometheus-Datenquelle sie automatisch anhängt.
7. Wählen Sie unter Auth den Schalter für SigV4-Authentifizierung aus, um sie zu aktivieren.

   Lassen Sie die Felder ARN-Rolle übernehmen und Externe ID leer. Wählen Sie dann für Standardregion die Region aus, in der sich Ihr Workspace in Amazon Managed Service für Prometheus befindet.
8. Wählen Sie Speichern & Testen aus.

   Sie sollten die folgende Nachricht sehen: Die Datenquelle funktioniert
9. Testen Sie eine PromQL-Abfrage anhand der neuen Datenquelle:

   a. Wählen Sie Explore.

   b. Führen Sie eine PromQL-Beispielabfrage aus, wie zum Beispiel:

   ```
   prometheus_tsdb_head_series
   ```

Abfragen mithilfe von APIs Prometheus-compatible

Obwohl die Verwendung eines Tools wie Amazon Managed Grafana die einfachste Methode ist, Ihre Metriken anzuzeigen und abzufragen, unterstützt Amazon Managed Service for Prometheus auch mehrere Prometheus-compatible APIs, mit denen Sie Ihre Metriken abfragen können. Weitere Informationen zu allen verfügbaren Prometheus-compatible APIs finden Sie unter. Prometheus-compatible APIs

Die Prometheus-compatible APIs verwenden die Prometheus-Abfragesprache PromQL, um die Daten anzugeben, die Sie zurückgeben möchten. Einzelheiten zu PromQL und seiner Syntax finden Sie unter Querying Prometheus in der Prometheus-Dokumentation.

Wenn Sie diese APIs verwenden, um Ihre Metriken abzufragen, müssen die Anfragen mit dem AWS Signature Version 4-Signaturprozess signiert werden. Sie können AWS Signature Version 4 einrichten, um den Signaturprozess zu vereinfachen. Weitere Informationen finden Sie unter aws-sigv4-proxy.

Das Signieren über AWS den SigV4-Proxy kann mit durchgeführt werden. `awscurl` Das folgende Thema Verwenden von awscurl zum Abfragen von Prometheus-compatible APIs führt Sie durch die Verwendung `awscurl` zur Einrichtung von SigV4. AWS

Themen

- Verwenden Sie awscurl für Abfragen mit APIs Prometheus-compatible

Verwenden Sie awscurl für Abfragen mit APIs Prometheus-compatible

API-Anfragen für Amazon Managed Service für Prometheus müssen mit SigV4 signiert werden. Sie können awscurl verwenden, um den Abfrageprozess zu vereinfachen.

Für die Installation von `awscurl` müssen Python 3 und der Pip-Package-Manager installiert sein.

Auf einer Linux-basierten Instance installiert der folgende Befehl `awscurl`.

```
$ pip3 install awscurl
```

Auf einem macOS-Computer installiert der folgende Befehl `awscurl`.

```
$ brew install awscurl
```

Das folgende Beispiel ist eine Beispielabfrage. `awscurl` Ersetzen Sie *`Region`* die *`QUERY`* Eingaben *`Workspace-id`* und durch entsprechende Werte für Ihren Anwendungsfall:

```
# 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**
>
> Ihre Abfragezeichenfolge muss URL-codiert sein.

Für eine Abfrage wie `query=up` könnten Sie Ergebnisse erhalten wie:

```
{
  "status": "success",
  "data": {
    "resultType": "vector",
    "result": [
      {
        "metric": {
          "__name__": "up",
          "instance": "localhost:9090",
          "job": "prometheus",
          "monitor": "monitor"
        },
        "value": [
          1652452637.636,
          "1"
        ]
      },
    ]
  }
}
```

Um die bereitgestellten Anfragen für `awscurl` zu signieren, müssen Sie die gültigen Anmeldeinformationen auf eine der folgenden Arten weitergeben:

- Geben Sie die Zugriffsschlüssel-ID und den geheimen Schlüssel für die IAM-Rolle an. Sie finden den Zugriffsschlüssel und den geheimen Schlüssel für die Rolle in der https://console.aws.amazon.com/iam/.

  Beispiel:

  ```
  $ 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>"
```

- Verweisen Sie auf die Konfigurationsdateien, die in der `.aws/credentials`und `/aws/config` Datei gespeichert sind. Sie können auch den Namen des zu verwendenden Profils angeben. Falls nicht angegeben, wird die `default` Datei verwendet. Beispiel:

```
$ 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>"
```

- Verwenden Sie das mit einer EC2-Instance verknüpfte Instance-Profil.

Ausführen von Abfrageanforderungen mithilfe des awscurl-Containers

Wenn die Installation einer anderen Version von Python und der zugehörigen Abhängigkeiten nicht möglich ist, kann ein Container verwendet werden, um die `awscurl` Anwendung und ihre Abhängigkeiten zu packen. Im folgenden Beispiel wird für die Bereitstellung eine Docker-Laufzeit verwendet`awscurl`, aber jede OCI-konforme Laufzeit und jedes Bild funktionieren.

```
$ 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"
```

Rufen Sie für jede Abfrage Statistiken über Ihre Abfragenutzung ab

Die Preisgestaltung für Abfragen basiert auf der Gesamtzahl der in einem Monat verarbeiteten Abfrageproben aus ausgeführten Abfragen. Sie können Statistiken zu jeder Anfrage abrufen, die Sie stellen, um den Überblick über Ihre verarbeiteten Proben zu behalten. Die Abfrageantwort für eine `query` oder eine `queryRange` API kann Statistikdaten zu Abfragebeispielen enthalten, die verarbeitet wurden, indem der Abfrageparameter `stats=all` in die Anfrage aufgenommen wird. In dem `samples` Objekt wird ein `stats` Objekt erstellt, und die `stats` Daten werden in der Antwort zurückgegeben.

Das `samples` Objekt hat die folgenden Attribute:

Attribut	Description
`totalQueryableSamples`	Gesamtzahl der verarbeiteten Abfragebeispiele. Dies sind die Informationen, die für die Abrechnung verwendet werden sollen.
`totalQueryableSamplesPerStep`	Die Anzahl der pro Schritt verarbeiteten Abfragebeispiele. Dies ist als Array von Arrays mit dem Zeitstempel in Epoche und der Anzahl der für den jeweiligen Schritt geladenen Beispiele strukturiert.

Nachfolgend finden Sie Beispiele für Anfragen und Antworten, die die `stats` Informationen in der Antwort enthalten:

Beispiel für `query`:

GET

```
endpoint/api/v1/query?query=up&time=1652382537&stats=all
```

Antwort

```
{
    "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
                    ]
                ]
            }
        }
    }
}
```

Beispiel für `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
```

Antwort

```
{
    "status": "success",
    "data": {
        "resultType": "matrix",
        "result": [
            {
                "metric": {},
                "values": [
                    [
                        1652383000,
                        "0"
```

```
                    ],
                    [
                        1652384000,
                        "0"
                    ]
                ]
            }
        ],
        "stats": {
            "samples": {
                "totalQueryableSamples": 8,
                "totalQueryableSamplesPerStep": [
                    [
                        1652382000,
                        0
                    ],
                    [
                        1652383000,
                        4
                    ],
                    [
                        1652384000,
                        4
                    ]
                ]
            }
        }
    }
}
```

Anomalie-Erkennung

Amazon Managed Service for Prometheus bietet Funktionen zur Erkennung von Anomalien, die mithilfe von Algorithmen für maschinelles Lernen automatisch ungewöhnliche Muster in Ihren Metrikdaten identifizieren. Diese Funktion hilft Ihnen dabei, proaktiv potenzielle Probleme zu erkennen, die Warnungsmüdigkeit zu verringern und Ihre Überwachungseffizienz zu verbessern, indem Sie sich auf wirklich anomales Verhalten statt auf statische Schwellenwerte konzentrieren.

Die Anomalieerkennung in Amazon Managed Service for Prometheus verwendet den Random Cut Forest (RCF) -Algorithmus, der Ihre Zeitreihendaten analysiert, um normale Verhaltensmuster zu ermitteln und Abweichungen von diesen Mustern zu identifizieren. Der Algorithmus passt sich an saisonale Trends an, behandelt fehlende Daten elegant und liefert Vertrauenswerte für erkannte Anomalien.

Funktionsweise der Anomalieerkennung

Amazon Managed Service for Prometheus verwendet maschinelles Lernen, um ungewöhnliche Muster in Metrikdaten ohne manuelle Schwellenwertkonfiguration zu identifizieren. Das System lernt normale Verhaltensmuster und saisonale Schwankungen kennen, wodurch Fehlalarme reduziert und Probleme frühzeitig erkannt werden können. Es passt sich kontinuierlich an Anwendungsänderungen an und eignet sich daher für dynamische Cloud-Umgebungen.

Die Anomalieerkennung überwacht Leistungskennzahlen von Anwendungen wie Reaktionszeiten und Fehlerraten, verfolgt den Zustand der Infrastruktur anhand der CPU- und Speicherauslastung, erkennt ungewöhnliches Benutzerverhalten, identifiziert anhand von Verkehrsanalysen den Bedarf an Kapazitätsplanung und überwacht Geschäftskennzahlen auf unerwartete Änderungen. Sie funktioniert am besten bei vorhersehbaren Mustern, saisonalen Schwankungen oder allmählichen Wachstumstrends.

Der RCF-Algorithmus (Random Cut Forest) wird zur Analyse von Zeitreihendaten verwendet. RCF erstellt Entscheidungsbäume, die den Datenraum unterteilen, und identifiziert isolierte Punkte, die weit von der Normalverteilung entfernt sind. Der Algorithmus lernt aus eingehenden Daten, um für jede Metrik ein dynamisches Modell des Normalverhaltens zu erstellen.

Wenn diese Option aktiviert ist, analysiert er historische Daten, um Basismuster und saisonale Trends zu ermitteln, generiert dann Prognosen für erwartete Werte und identifiziert Abweichungen. Der Algorithmus erzeugt vier wichtige Ergebnisse:

- upper_band — Die obere Grenze der erwarteten Normalwerte
- lower_band — Die untere Grenze der erwarteten Normalwerte
- score — Ein numerischer Anomaliewert, der angibt, wie ungewöhnlich der Datenpunkt ist
- Wert — Der tatsächlich beobachtete Metrikwert

Erste Schritte mit der Anomalieerkennung

Um mit der Verwendung der Anomalieerkennung mit Ihren Prometheus-Metriken zu beginnen, benötigen Sie genügend historische Daten, damit der Algorithmus normale Muster erkennen kann. Für optimale Ergebnisse empfehlen wir, mindestens 14 Tage über konsistente Metrikdaten zu verfügen, bevor Sie die Anomalieerkennung aktivieren.

Mithilfe der API können Sie in einer Vorschau sehen, wie die Anomalieerkennung mit Ihren Metriken funktionieren wird. `PreviewAnomalyDetector` Verwenden Sie diese Option`PreviewAnomalyDetector`, um den Algorithmus anhand Ihrer historischen Daten zu testen und seine Wirksamkeit zu bewerten, bevor Sie ihn in die Produktionsüberwachung implementieren. Weitere Informationen finden Sie unter PreviewAnomalyDetector API.

Beachten Sie bei der Implementierung der Anomalieerkennung die folgenden bewährten Methoden:

- Beginnen Sie mit stabilen Kennzahlen — Beginnen Sie mit Kennzahlen, die konsistente Muster aufweisen, und vermeiden Sie zunächst sehr volatile oder spärliche Daten.
- Verwenden Sie aggregierte Daten — Wenden Sie die Anomalieerkennung auf aggregierte Kennzahlen (wie Durchschnittswerte oder Summen) an und nicht auf Rohdaten mit hoher Kardinalität, um eine bessere Leistung und Genauigkeit zu erzielen.
- Sensitivität optimieren — Passen Sie die Algorithmusparameter an Ihren spezifischen Anwendungsfall und Ihre Toleranz für Fehlalarme im Vergleich zu übersehenen Anomalien an.
- Überwachen Sie die Leistung des Algorithmus — Überprüfen Sie regelmäßig festgestellte Anomalien, um sicherzustellen, dass der Algorithmus auch bei der Weiterentwicklung Ihres Systems wertvolle Erkenntnisse liefert.

PreviewAnomalyDetector API

Verwenden Sie den `PreviewAnomalyDetector` Vorgang, um einen Endpunkt zu erstellen, der demonstriert, wie Ihre Metrikdaten durch den Algorithmus zur Erkennung von Anomalien während

des angegebenen Zeitraums analysiert werden. Dieser Endpunkt hilft Ihnen, die Leistung des Detektors vor der Implementierung zu bewerten und zu validieren.

Gültige HTTP-Verben

`GET`, `POST`

Unterstützte Payload-Typen

URL-kodierte Parameter

`application/x-www-form-urlencoded` für `POST`

Unterstützte Parameter

`query=<string>`Ein Abfrage-String mit einem Prometheus-Ausdruck.

`start=<rfc3339 | unix_timestamp>` Startzeitstempel, wenn Sie `query_range` einen bestimmten Zeitraum abfragen möchten.

`end=<rfc3339 | unix_timestamp>` Endzeitstempel, wenn Sie für `query_range` die Abfrage nach einem bestimmten Zeitraum verwenden.

`step=<duration | float>` Schrittweite der Auflösung im `duration` Format oder als `float` Anzahl von Sekunden abfragen. Verwenden Sie diese Option nur, wenn Sie `query_range` für Abfragen für einen bestimmten Zeitraum verwenden und dies für solche Abfragen erforderlich sind.

Formatierung von Abfrageparametern

Verpacken Sie Ihren ursprünglichen PromQL-Ausdruck mit der Pseudofunktion RandomCutForest (RCF) im Abfrageparameter. Weitere Informationen finden Sie RandomCutForestConfigurationin der Amazon Managed Service for Prometheus API-Referenz.

Die RCF-Funktion verwendet dieses Format:

```
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]]]]])
```

Alle Parameter außer der Abfrage sind optional und verwenden Standardwerte, wenn sie weggelassen werden. Die minimale Syntax lautet:

```
RCF(<query>)
```

Sie müssen Ihre Abfrage mit einer Aggregationsfunktion umschließen. Wenn Sie bestimmte optionale Parameter verwenden und andere weglassen möchten, lassen Sie leere Positionen in der Funktion übrig:

```
RCF(<query>,,,,,1.0,1.0)
```

In diesem Beispiel werden nur die Verhältnisparameter festgelegt, die Spitzen und Abnahmen bei der Erkennung von Anomalien auf der Grundlage des Verhältnisses zwischen erwarteten und beobachteten Werten ignorieren.

API-Anforderung und -Antwort

Erfolgreiche Aufrufe geben dasselbe Format wie die QueryMetrics API zurück. Zusätzlich zu den ursprünglichen Zeitreihen gibt die API diese neuen Zeitreihen zurück, wenn genügend Stichproben verfügbar sind:

- `anomaly_detector_preview:lower_band`— Unteres Band für den erwarteten Wert des PromQL-Ausdrucksergebnisses
- `anomaly_detector_preview:score`— Anomalie-Score zwischen 0 und 1, wobei 1 auf eine hohe Zuverlässigkeit einer Anomalie an diesem Datenpunkt hinweist
- `anomaly_detector_preview:upper_band`— Oberes Band für den erwarteten Wert des PromQL-Ausdrucksergebnisses

Beispielanforderung

```
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
```

Beispielantwort

```
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"
  }
}
```

Verwenden von Regeln zum Ändern oder Überwachen von Metriken, sobald sie empfangen werden

Sie können Regeln einrichten, um auf Metriken zu reagieren, sobald sie von Amazon Managed Service for Prometheus empfangen werden. Diese Regeln können die Metriken überwachen oder sogar neue, berechnete Metriken auf der Grundlage der erhaltenen Metriken erstellen.

Amazon Managed Service für Prometheus unterstützt zwei Arten von Regeln, die in regelmäßigen Abständen ausgewertet werden:

- Mit Aufzeichnungsregeln können Sie häufig benötigte oder rechenintensive Ausdrücke vorab berechnen und deren Ergebnisse als neue Zeitreihen speichern. Das Abfragen des vorberechneten Ergebnisses ist oft viel schneller, als den ursprünglichen Ausdruck jedes Mal auszuführen, wenn er benötigt wird.
- Mit Alarmregeln können Sie Alarmbedingungen auf der Grundlage von PromQL und einem Schwellenwert definieren. Wenn die Regel den Schwellenwert auslöst, wird eine Benachrichtigung an den Alert Manager gesendet, der so konfiguriert werden kann, dass er die Regeln verwaltet oder sie an eine nachgeschaltete Benachrichtigung an Empfänger wie Amazon Simple Notification Service weiterleitet.

Um Regeln in Amazon Managed Service für Prometheus zu verwenden, erstellen Sie eine oder mehrere YAML-Regeldateien, die die Regeln definieren. Eine Regeldatei für Amazon Managed Service für Prometheus hat dasselbe Format wie eine Regeldatei in der eigenständigen Version von Prometheus. Weitere Informationen finden Sie in der Prometheus-Dokumentation unter Definieren von Aufzeichnungsregeln und Alarmregeln.

Sie können mehrere Regeldateien in einem Workspace haben. Jede einzelne Regeldatei ist in einem separaten Namespace enthalten. Wenn Sie mehrere Regeldateien haben, können Sie bestehende Prometheus-Regeldateien in einen Workspace importieren, ohne sie ändern oder kombinieren zu müssen. Verschiedene Regelgruppen-Namespaces können auch unterschiedliche Tags haben.

Sequenzierung von Regeln

In einer Regeldatei sind Regeln in Regelgruppen enthalten. Regeln innerhalb einer einzelnen Regelgruppe in einer Regeldatei werden immer in der Reihenfolge von oben nach unten ausgewertet. Daher kann in Aufzeichnungsregeln das Ergebnis einer Aufzeichnungsregel bei der Berechnung einer späteren Aufzeichnungsregel oder in einer Alarmregel in derselben Regelgruppe verwendet

werden. Da Sie jedoch nicht die Reihenfolge angeben können, in der separate Regeldateien ausgeführt werden, können Sie die Ergebnisse einer Aufzeichnungsregel nicht verwenden, um eine Regel in einer anderen Regelgruppe oder einer anderen Regeldatei zu berechnen.

Themen

- Grundlegendes zu den IAM-Berechtigungen, die für die Verwendung von Regeln erforderlich sind
- Erstellen Sie eine Regeldatei
- Laden Sie eine Regelkonfigurationsdatei auf Amazon Managed Service for Prometheus hoch
- Bearbeiten oder ersetzen Sie eine Regelkonfigurationsdatei
- Problembehandlung bei Regelauswertungen
- Fehlersuche-Ruler

Grundlegendes zu den IAM-Berechtigungen, die für die Verwendung von Regeln erforderlich sind

Sie müssen Benutzern Berechtigungen zur Verwendung von Regeln in Amazon Managed Service für Prometheus erteilen. Erstellen Sie eine AWS Identity and Access Management (IAM-) Richtlinie mit den folgenden Berechtigungen und weisen Sie die Richtlinie Ihren Benutzern, Gruppen oder Rollen zu.

> **Note**
>
> Weitere Informationen zu IAM finden Sie unter Identitäts- und Zugriffsverwaltung für Amazon Managed Service für Prometheus.

Richtlinie zur Gewährung des Zugriffs auf Nutzungsregeln

Die folgende Richtlinie gewährt Zugriff auf die Nutzungsregeln für alle Ressourcen in Ihrem Konto.

JSON

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

```
            "Effect": "Allow",
            "Action": [
                "aps:CreateRuleGroupsNamespace",
                "aps:ListRuleGroupsNamespaces",
                "aps:DescribeRuleGroupsNamespace",
                "aps:PutRuleGroupsNamespace",
                "aps:DeleteRuleGroupsNamespace"
            ],
            "Resource": "*"
        }
    ]
}
```

Richtlinie, um nur Zugriff auf einen Namespace zu gewähren

Sie können auch eine Richtlinie erstellen, die nur Zugriff auf bestimmte Richtlinien gewährt. Die folgende Beispielrichtlinie gewährt nur Zugriff auf die `RuleGroupNamespace` angegebenen. Um diese Richtlinie zu verwenden, ersetzen Sie*<account>*, *<region><workspace-id>*, und *<namespace-name>* durch die entsprechenden Werte für Ihr Konto.

Erstellen Sie eine Regeldatei

Um Regeln in Amazon Managed Service für Prometheus zu verwenden, erstellen Sie eine Regeldatei, die die Regeln definiert. Eine Regeldatei für Amazon Managed Service for Prometheus ist eine YAML-Textdatei, die dasselbe Format wie eine Regeldatei in der eigenständigen Version von Prometheus hat. Weitere Informationen finden Sie in der Prometheus-Dokumentation unter Definieren von Aufzeichnungsregeln und Warnregeln.

Im Folgenden wird ein Basisbeispiel für eine Regeldatei dargestellt:

```
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"
```

In diesem Beispiel wird eine Regelgruppe erstelltcpu_metrics, die alle 60 Sekunden ausgewertet wird. Diese Regelgruppe erstellt mithilfe einer Aufzeichnungsregel eine neue Metrik, die aufgerufen wird, avg_cpu_usage und verwendet diese dann in einer Warnung. Im Folgenden werden einige der verwendeten Eigenschaften beschrieben. Weitere Informationen zu Warnregeln und anderen Eigenschaften, die Sie einbeziehen können, finden Sie unter Warnungsregeln in der Prometheus-Dokumentation.

- `record: avg_cpu_usage`— Diese Aufzeichnungsregel erstellt eine neue Metrik namens. `avg_cpu_usage`
- Das Standardauswertungsintervall für Regelgruppen beträgt 60 Sekunden, wenn die `interval` Eigenschaft nicht angegeben ist.
- `expr: avg(rate(node_cpu_seconds_total[5m])) by (instance)`— Dieser Ausdruck für die Aufzeichnungsregel berechnet die durchschnittliche CPU-Auslastung der letzten 5 Minuten für jeden Knoten, gruppiert nach der `instance` Bezeichnung.
- `alert: HighAverageCPU`— Diese Warnungsregel erstellt eine neue Warnung namens `HighAverageCPU`
- `expr: avg_cpu_usage > 0.8` — Dieser Ausdruck weist die Warnung an, nach Stichproben zu suchen, bei denen die durchschnittliche CPU-Auslastung über 80% liegt.
- `for: 10m`— Die Warnung wird nur ausgelöst, wenn die durchschnittliche CPU-Auslastung mindestens 10 Minuten lang 80% übersteigt.

  In diesem Fall wird die Metrik als Durchschnitt der letzten 5 Minuten berechnet. Die Warnung wird also nur ausgelöst, wenn es mindestens zwei aufeinanderfolgende 5-minütige Stichproben (insgesamt 10 Minuten) gibt, bei denen die durchschnittliche CPU-Auslastung über 80% liegt.
- `keep_firing_for: 20m`— Diese Warnung wird so lange ausgelöst, bis die Samples mindestens 20 Minuten lang unter dem Schwellenwert liegen. Dies kann nützlich sein, um zu verhindern, dass der Alarm wiederholt nacheinander hoch und runter geht.

> **Note**
>
> Sie können lokal eine Regeldefinitionsdatei erstellen und sie dann auf Amazon Managed Service for Prometheus hochladen, oder Sie können die Definition direkt in der Amazon

Managed Service for Prometheus-Konsole erstellen, bearbeiten und hochladen. In beiden Fällen gelten dieselben Formatierungsregeln. Weitere Informationen zum Hochladen und Bearbeiten Ihrer Datei finden Sie unterLaden Sie eine Regelkonfigurationsdatei auf Amazon Managed Service for Prometheus hoch.

Laden Sie eine Regelkonfigurationsdatei auf Amazon Managed Service for Prometheus hoch

Sobald Sie wissen, welche Regeln Sie in Ihrer Regelkonfigurationsdatei haben möchten, können Sie sie entweder in der Konsole erstellen und bearbeiten, oder Sie können eine Datei mit der Konsole hochladen oder. AWS CLI

> **Note**
>
> Wenn Sie einen Amazon EKS-Cluster ausführen, können Sie mit AWS Controllers for Kubernetes auch eine Regelkonfigurationsdatei hochladen.

So verwenden Sie die Amazon Managed Service for Prometheus-Konsole, um Ihre Regelkonfiguration zu bearbeiten oder zu ersetzen und den Namespace zu erstellen

1. Öffnen Sie die Amazon Managed Service for Prometheus-Konsole unter. https://console.aws.amazon.com/prometheus/
2. Wählen Sie in der oberen linken Ecke der Seite das Menüsymbol und dann Alle Workspaces aus.
3. Wählen Sie die Workspace-ID des Workspace und dann den Tab Regelverwaltung aus.
4. Wählen Sie Namespace hinzufügen.
5. Wählen Sie Datei auswählen und wählen Sie die Regeldefinitionsdatei aus.

   Alternativ können Sie eine Regeldefinitionsdatei direkt in der Amazon Managed Service for Prometheus-Konsole erstellen und bearbeiten, indem Sie Konfiguration definieren auswählen. Dadurch wird eine Muster-Standarddefinitionsdatei erstellt, die Sie vor dem Hochladen bearbeiten.
6. (Optional) Zum Hinzufügen von Tags zum Namespace wählen Sie Neues Tag hinzufügen aus.

Geben Sie für Key (Schlüssel) einen Namen für das Tag ein. Sie können einen optionalen Wert für das Tag unter Value (Wert) hinzufügen.

Zum Hinzufügen eines weiteren Tags wählen Sie Neues Tag hinzufügen erneut aus.

7. Klicken Sie auf Weiter. Amazon Managed Service für Prometheus erstellt einen neuen Namespace mit demselben Namen wie die von Ihnen ausgewählte Regeldatei.

Um das AWS CLI um eine Alert Manager-Konfiguration in einen Workspace in einem neuen Namespace hochzuladen

1. Kodiert den Inhalt Ihrer Alert-Manager-Datei mit Base64. Sie können für Linux den folgenden Befehl verwenden:

```
base64 input-file output-file
```

Sie können für macOS den folgenden Befehl verwenden:

```
openssl base64 input-file output-file
```

2. Geben Sie einen der folgenden Befehle ein, um den Namespace zu erstellen und die Datei hochzuladen.

Geben Sie in AWS CLI Version 2 Folgendes ein:

```
aws amp create-rule-groups-namespace --data file://path_to_base_64_output_file --
name namespace-name  --workspace-id my-workspace-id --region region
```

Geben Sie in AWS CLI Version 1 Folgendes ein:

```
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. Es dauert einige Sekunden, bis Ihre Alert-Manager-Konfiguration aktiv wird. Geben Sie zur Prüfung des Status den folgenden Befehl ein:

```
aws amp describe-rule-groups-namespace --workspace-id workspace_id --
name namespace-name --region region
```

Ist `status` `ACTIVE`, ist Ihre Regeldatei wirksam geworden.

Bearbeiten oder ersetzen Sie eine Regelkonfigurationsdatei

Wenn Sie die Regeln in einer Regeldatei ändern möchten, die Sie bereits in Amazon Managed Service for Prometheus hochgeladen haben, können Sie entweder eine neue Regeldatei hochladen, um die bestehende Konfiguration zu ersetzen, oder Sie können die aktuelle Konfiguration direkt in der Konsole bearbeiten. Optional können Sie die aktuelle Datei herunterladen, sie in einem Texteditor bearbeiten und dann die neue Version hochladen.

Verwenden Sie die Konsole von Amazon Managed Service für Prometheus wie folgt, um Ihre Regelkonfiguration zu bearbeiten

1. Öffnen Sie die Amazon Managed Service for Prometheus-Konsole unter. https://console.aws.amazon.com/prometheus/
2. Wählen Sie in der oberen linken Ecke der Seite das Menüsymbol und dann Alle Workspaces aus.
3. Wählen Sie die Workspace-ID des Workspace und dann den Tab Regelverwaltung aus.
4. Wählen Sie den Namen der Regelkonfigurationsdatei aus, die Sie bearbeiten möchten.
5. (Optional) Wenn Sie die aktuelle Regelkonfigurationsdatei herunterladen möchten, wählen Sie Herunterladen oder Kopieren.
6. Wählen Sie Ändern, um die Konfiguration direkt in der Konsole zu bearbeiten. Wählen Sie „Speichern“, wenn Sie fertig sind.

   Alternativ können Sie Konfiguration ersetzen wählen, um eine neue Konfigurationsdatei hochzuladen. Wenn ja, wählen Sie die neue Regeldefinitionsdatei aus und klicken Sie auf Weiter, um sie hochzuladen.

Um das AWS CLI um eine Regelkonfigurationsdatei zu bearbeiten

1. Base64 kodiert den Inhalt Ihrer Regeldatei. Sie können für Linux den folgenden Befehl verwenden:

   ```
   base64 input-file output-file
   ```

Sie können für macOS den folgenden Befehl verwenden:

```
openssl base64 input-file output-file
```

2. Geben Sie einen der folgenden Befehle ein, um die neue Datei hochzuladen.

   Geben Sie in AWS CLI Version 2 Folgendes ein:

   ```
   aws amp put-rule-groups-namespace --data file://path_to_base_64_output_file --
   name namespace-name  --workspace-id my-workspace-id --region region
   ```

   Geben Sie in AWS CLI Version 1 Folgendes ein:

   ```
   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. Es dauert einige Sekunden, bis Ihre Regeldatei aktiv wird. Geben Sie zur Prüfung des Status den folgenden Befehl ein:

   ```
   aws amp describe-rule-groups-namespace --workspace-id workspace_id --
   name namespace-name --region region
   ```

   Ist `status` `ACTIVE`, ist Ihre Regeldatei wirksam geworden. Bis dahin ist die vorherige Version dieser Regeldatei noch aktiv.

Problembehandlung bei Regelauswertungen

Dieses Handbuch enthält schrittweise Verfahren zur Fehlerbehebung für häufig auftretende Probleme mit Regelauswertungen in Amazon Managed Service for Prometheus (AMP). Folgen Sie diesen Verfahren, um Probleme mit Ihren Warn- und Aufzeichnungsregeln zu diagnostizieren und zu lösen.

Themen

- Überprüfen Sie den Status der Alarmauslösung
- Beheben Sie fehlende Warnmeldungen
- Überprüfen Sie den Integritätsstatus der Regel
- Verwenden Sie Offset in Abfragen, um Verzögerungen bei der Datenerfassung zu beheben
- Häufige Probleme und Lösungen

- Bewährte Verfahren für die Bewertung von Regeln

Überprüfen Sie den Status der Alarmauslösung

Überprüfen Sie bei der Behebung von Problemen mit der Regelauswertung zunächst, ob Ihre Warnung ausgelöst wurde, indem Sie die synthetische Zeitreihe `ALERTS` abfragen. Die `ALERTS` Zeitreihen enthalten die folgenden Bezeichnungen:

- alertname — Der Name der Warnung.
- alertstate — Entweder ausstehend oder wird ausgelöst.
  - ausstehend — Die Warnung wartet auf die in der Klausel angegebene Dauer. `for`
  - Auslösen — Die Warnung hat die Bedingungen für die angegebene Dauer erfüllt. Zusätzliche Bezeichnungen sind in Ihrer Warnregel definiert.

> **Note**
>
> Solange eine Warnung ausgelöst wird oder aussteht, ist der Beispielwert 1. Wenn Ihre Warnung inaktiv ist, werden keine Proben produziert.

Beheben Sie fehlende Warnmeldungen

Wenn Benachrichtigungen ausgelöst werden, aber keine Benachrichtigungen eintreffen, überprüfen Sie die folgenden Alertmanager-Einstellungen:

1. Überprüfen Sie Ihre Alertmanager-Konfiguration — Vergewissern Sie sich, dass Routen, Empfänger und Einstellungen korrekt konfiguriert sind. Überprüfen Sie die Einstellungen für die Routenblockierung, einschließlich Wartezeiten, Zeitintervalle und erforderlicher Beschriftungen, die sich auf die Auslösung von Alarmen auswirken können. Vergleichen Sie die Alarmregeln mit den entsprechenden Routen und Empfängern, um sicherzustellen, dass sie korrekt übereinstimmen. Stellen Sie bei Routen mit sicher`time_interval`, dass die Zeitstempel innerhalb der angegebenen Intervalle liegen.
2. Überprüfen Sie die Berechtigungen des Alarmempfängers — Wenn Sie ein Amazon SNS SNS-Thema verwenden, stellen Sie sicher, dass AMP über die erforderlichen Berechtigungen zum Veröffentlichen von Benachrichtigungen verfügt. Weitere Informationen finden Sie unter Amazon

Managed Service for Prometheus die Erlaubnis erteilen, Warnmeldungen zu Ihrem Amazon SNS SNS-Thema zu senden.

3. Überprüfen Sie die Payload-Kompatibilität des Empfängers — Vergewissern Sie sich, dass Ihr Alert-Empfänger das Payload-Format von Alertmanager akzeptiert. Informationen zu den Amazon SNS SNS-Anforderungen finden Sie unterGrundlegendes zu den Amazon SNS SNS-Nachrichtenvalidierungsregeln.
4. Überprüfen Sie die Alertmanager-Protokolle — AMP stellt von Alertmanager bereitgestellte Protokolle zur Verfügung, um bei der Behebung von Benachrichtigungsproblemen zu helfen. Weitere Informationen finden Sie unter Überwachen Sie Amazon Managed Service für Prometheus-Ereignisse mit Protokollen CloudWatch.

Weitere Informationen zu Alertmanager finden Sie unter. Verwaltung und Weiterleitung von Benachrichtigungen in Amazon Managed Service für Prometheus mit Alert Manager

Überprüfen Sie den Integritätsstatus der Regel

Falsch formatierte Regeln können zu Fehlern bei der Evaluierung führen. Verwenden Sie die folgenden Methoden, um zu ermitteln, warum eine Regel nicht ausgewertet werden konnte:

Example

Verwenden Sie die ListRules API

Die ListRules API bietet Informationen zur Regelintegrität. Überprüfen Sie die `lastError` Felder `health` und, um Probleme zu diagnostizieren.

Beispiel für eine Antwort:

```
{
  "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

Verwenden Sie verkaufte Logs

Die ListRules API zeigt nur die neuesten Informationen an. Für einen detaillierteren Verlauf aktivierst du in deinem Workspace die Option „Verkaufte Logs“ für den Zugriff auf:

- Zeitstempel von Fehlschlägen bei der Evaluierung
- Detaillierte Fehlermeldungen
- Historische Bewertungsdaten

Beispiel für eine versendete Protokollnachricht:

```
{
  "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"
}
```

Weitere Beispiele für Logs von Ruler oder Alertmanager finden Sie unter und. Fehlersuche-Ruler Verwaltung und Weiterleitung von Benachrichtigungen in Amazon Managed Service für Prometheus mit Alert Manager

Verwenden Sie Offset in Abfragen, um Verzögerungen bei der Datenerfassung zu beheben

Standardmäßig werden Ausdrücke ohne Offset (Sofortabfrage) ausgewertet, wobei die Werte zum Zeitpunkt der Auswertung verwendet werden. Wenn die Erfassung von Metriken verzögert erfolgt, stellen die Aufzeichnungsregeln möglicherweise nicht dieselben Werte dar wie bei der manuellen Auswertung des Ausdrucks, nachdem alle Metriken aufgenommen wurden.

> **Tip**
>
> Sie können einen Linealabfrageversatz auf Workspace-Ebene so konfigurieren, dass eine globale Verzögerung auf alle Regelauswertungen angewendet wird, ohne einzelne Regelausdrücke zu ändern. Weitere Informationen finden Sie unter Konfiguriere deinen Workspace. Alternativ können Sie den PromQL-Offset-Modifikator für einzelne Ausdrücke verwenden. Weitere Informationen finden Sie unter Offset-Modifikator in der Prometheus-Dokumentation.

Beispiel: Umgang mit verzögerten Metriken

Wenn Ihre Regel um 12:00 Uhr ausgewertet wird, die letzte Stichprobe für die Metrik aber aufgrund der Verzögerung bei der Datenaufnahme von 11:45 Uhr stammt, findet die Regel bis 12:00 Uhr keine Stichproben. Um dies zu verringern, fügen Sie einen Offset hinzu, z. B.:. **`my_metric_name offset 15m`**

Beispiel: Behandeln Sie Metriken aus mehreren Quellen

Wenn Metriken aus verschiedenen Quellen stammen, z. B. aus zwei Servern, können sie zu unterschiedlichen Zeiten aufgenommen werden. Um dem entgegenzuwirken, formulieren Sie einen Ausdruck wie: **`metric_from_server_A / metric_from_server_B`**

Wenn die Regel zwischen den Aufnahmezeiten von Server A und Server B ausgewertet wird, erhalten Sie unerwartete Ergebnisse. Die Verwendung eines Offsets kann dazu beitragen, die Evaluierungszeiten aufeinander abzustimmen.

Häufige Probleme und Lösungen

Lücken bei der Aufzeichnung von Regeldaten

Wenn Sie Lücken in den Daten Ihrer Aufzeichnungsregeln im Vergleich zur manuellen Auswertung feststellen (wenn Sie den ursprünglichen PromQL-Ausdruck der Aufzeichnungsregel direkt über die Abfrage-API oder Benutzeroberfläche ausführen), kann dies auf einen der folgenden Gründe zurückzuführen sein:

1. Lange Bewertungszeiten — Eine Regelgruppe kann nicht mehrere gleichzeitige Auswertungen haben. Wenn die Evaluierungszeit das konfigurierte Intervall überschreitet, können nachfolgende Evaluierungen verpasst werden. Mehrere aufeinanderfolgende verpasste Bewertungen, die das konfigurierte Intervall überschreiten, können dazu führen, dass die Aufzeichnungsregel veraltet ist. Weitere Informationen finden Sie unter Staleness in der Prometheus-Dokumentation. Sie können die Testdauer anhand der CloudWatch Metrik überwachen`RuleGroupLastEvaluationDuration`, um Regelgruppen zu identifizieren, deren Auswertung zu lange dauert.
2. Überwachung verpasster Bewertungen — AMP bietet die `RuleGroupIterationsMissed` CloudWatch Metrik, mit der Sie nachverfolgen können, wann Bewertungen übersprungen wurden. Die ListRules API zeigt für jede Bewertung den Zeitpunkt der Bewertung und den Zeitpunkt der letzten Bewertung an rule/group, was dazu beitragen kann, Muster verpasster Bewertungen zu identifizieren. Weitere Informationen finden Sie unter ListRules.

Empfehlung: Teilen Sie die Regeln in separate Gruppen auf

Um die Dauer der Evaluierung zu verkürzen, teilen Sie die Regeln in separate Regelgruppen auf. Regeln innerhalb einer Gruppe werden sequentiell ausgeführt, während Regelgruppen parallel ausgeführt werden können. Bewahren Sie verwandte Regeln, die voneinander abhängen, in derselben Gruppe auf. Im Allgemeinen sorgen kleinere Regelgruppen für einheitlichere Bewertungen und weniger Lücken.

Bewährte Verfahren für die Bewertung von Regeln

1. Optimieren Sie die Größe der Regelgruppen — Halten Sie die Regelgruppen klein, um konsistente Bewertungen zu gewährleisten. Gruppieren Sie verwandte Regeln zusammen, vermeiden Sie jedoch große Regelgruppen.
2. Legen Sie angemessene Bewertungsintervalle fest — sorgen Sie für ein ausgewogenes Verhältnis zwischen rechtzeitigen Warnmeldungen und Systemlast. Überprüfen Sie die Stabilitätsmuster Ihrer überwachten Messwerte, um deren normale Schwankungsbreite zu ermitteln.
3. Verwenden Sie Offset-Modifikatoren für verspätete Messwerte — Fügen Sie Offsets hinzu, um Verzögerungen bei der Datenerfassung auszugleichen. Passen Sie die Offset-Dauer anhand der beobachteten Aufnahmemuster an.
4. Überwachen Sie die Evaluierungsleistung — Verfolgen Sie die Metrik`RuleGroupIterationsMissed`. Überprüfen Sie die Testzeiten in der ListRules API.
5. Regelausdrücke validieren — Stellen Sie sicher, dass die Ausdrücke zwischen Regeldefinitionen und manuellen Abfragen exakt übereinstimmen. Testen Sie Ausdrücke mit unterschiedlichen Zeiträumen, um das Verhalten zu verstehen.
6. Regelmäßige Überprüfung der Regelintegrität — Prüfen Sie die Regelauswertung auf Fehler. Überwachen Sie die ausgelieferten Protokolle auf wiederkehrende Probleme.

Wenn Sie diese Schritte und bewährten Methoden zur Fehlerbehebung befolgen, können Sie häufig auftretende Probleme mit Regelauswertungen in Amazon Managed Service for Prometheus identifizieren und lösen.

Fehlersuche-Ruler

Mit Überwachen Sie Amazon Managed Service für Prometheus-Ereignisse mit Protokollen CloudWatch können Sie Probleme im Zusammenhang mit Alert Manager und Ruler beheben. Dieser Abschnitt enthält Themen zur Fehlerbehebung im Zusammenhang mit Ruler.

Wenn das Protokoll den folgenden Ruler-Fehler enthält

```
{
    "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"
}
```

Dies bedeutet, dass bei der Ausführung der Regel ein Fehler aufgetreten ist.

Maßnahme

Verwenden Sie die Fehlermeldung, um Fehler bei der Regelausführung zu beheben.

Verwaltung und Weiterleitung von Benachrichtigungen in Amazon Managed Service für Prometheus mit Alert Manager

Wenn die von Amazon Managed Service für Prometheus ausgeführten Alarmregeln ausgelöst werden, verarbeitet der Alert Manager die gesendeten Alarme. Es dedupliziert, gruppiert und leitet die Warnmeldungen an nachgeschaltete Empfänger weiter. Amazon Managed Service für Prometheus unterstützt nur Amazon Simple Notification Service als Empfänger und kann Nachrichten an Amazon-SNS-Themen im selben Konto weiterleiten. Sie können den Alert Manager auch verwenden, um Alarme stummzuschalten und zu unterdrücken.

Alert Manager bietet ähnliche Funktionen wie Alertmanager in Prometheus.

Sie können die Konfigurationsdatei des Alert Managers für Folgendes verwenden:

- Gruppierung – Bei der Gruppierung werden ähnliche Alarme in einer einzigen Benachrichtigung zusammengefasst. Dies ist besonders nützlich bei größeren Ausfällen, wenn viele Systeme gleichzeitig ausfallen und Hunderte von Alarme gleichzeitig ausgelöst werden können. Angenommen, ein Netzwerkausfall führt dazu, dass viele Ihrer Knoten gleichzeitig ausfallen. Wenn diese Arten von Alarme gruppiert sind, sendet Ihnen der Alert Manager eine einzige Benachrichtigung.

  Die Gruppierung von Alarme und der Zeitpunkt der gruppierten Benachrichtigungen werden durch einen Routing-Baum in der Alert-Manager-Konfigurationsdatei konfiguriert. Weitere Informationen finden Sie unter <Route>.
- Unterdrückung – Durch die Unterdrückung werden Benachrichtigungen für bestimmte Alarme unterbunden, wenn bestimmte andere Alarme bereits ausgelöst werden. Wenn beispielsweise ein Alarm ausgelöst wird, dass ein Cluster nicht erreichbar ist, können Sie den Alert Manager so konfigurieren, dass alle anderen Alarme, die diesen Cluster betreffen, stummgeschaltet werden. Dadurch werden Benachrichtigungen für Hunderte oder Tausende von Alarme verhindert, die nichts mit dem eigentlichen Problem zu tun haben. Weitere Informationen zum Schreiben von Unterdrückungsregeln finden Sie unter <inhibit_rule>.
- Stummschalten – Schaltet Alarme für eine bestimmte Zeit stumm, z. B. während eines Wartungsfensters. Eingehende Alarme werden daraufhin überprüft, ob sie mit allen Gleichheits- oder regulären Ausdrucksübereinstimmungen einer aktiven Stummschaltung übereinstimmen. Wenn dies der Fall ist, werden keine Benachrichtigungen für diesen Alarm gesendet.

Um eine Stummschaltung zu erstellen, verwenden Sie die `PutAlertManagerSilences` API. Weitere Informationen finden Sie unter PutAlertManagerSilences.

Prometheus Vorlagensystem

Standalone Prometheus unterstützt Templating unter Verwendung separater Vorlagendateien. Vorlagen können unter anderem Bedingungen verwenden und Daten formatieren.

In Amazon Managed Service for Prometheus fügen Sie Ihr Template in dieselbe Alert Manager-Konfigurationsdatei ein wie Ihre Alert Manager-Konfiguration.

Themen

- Grundlegendes zu den IAM-Berechtigungen, die für die Arbeit mit dem Alert Manager erforderlich sind
- Erstellen Sie in Amazon Managed Service eine Alert Manager-Konfiguration für Prometheus, um Benachrichtigungen zu verwalten und weiterzuleiten
- Leiten Sie Benachrichtigungen mit dem Alert Manager in Amazon Managed Service für Prometheus an einen Alertempfänger weiter
- Laden Sie Ihre Alert Manager-Konfigurationsdatei auf Amazon Managed Service for Prometheus hoch
- Integrieren Sie Benachrichtigungen in Amazon Managed Grafana oder Open-Source-Grafana
- Beheben Sie Fehler im Alert Manager mithilfe von CloudWatch Protokollen

Grundlegendes zu den IAM-Berechtigungen, die für die Arbeit mit dem Alert Manager erforderlich sind

Sie müssen Benutzern Berechtigungen zur Verwendung des Alert Managers in Amazon Managed Service for Prometheus erteilen. Erstellen Sie eine AWS Identity and Access Management (IAM)-Richtlinie mit den folgenden Berechtigungen und weisen Sie die Richtlinie Ihren Benutzern, Gruppen oder Rollen zu.

Erstellen Sie in Amazon Managed Service eine Alert Manager-Konfiguration für Prometheus, um Benachrichtigungen zu verwalten und weiterzuleiten

Um den Alert Manager und die Vorlage in Amazon Managed Service für Prometheus zu verwenden, erstellen Sie eine YAML-Konfigurationsdatei für den Alert Manager. Eine Alert-Manager-Datei für Amazon Managed Service für Prometheus besteht aus zwei Hauptabschnitten:

- `template_files:` enthält die Vorlagen für Nachrichten, die von Empfängern gesendet werden. Weitere Informationen finden Sie in der Prometheus-Dokumentation unter Vorlagenreferenz und Vorlagenbeispiele.
- `alertmanager_config:` enthält die Alert-Manager-Konfiguration. Dies verwendet dieselbe Struktur wie eine Alert-Manager-Konfigurationsdatei im eigenständigen Prometheus. Weitere Informationen finden Sie unter Konfiguration in der Alertmanager-Dokumentation.

> **Note**
>
> Die in der obigen Prometheus-Dokumentation beschriebene `repeat_interval`-Konfiguration hat eine zusätzliche Einschränkung in Amazon Managed Service für Prometheus. Der maximal zulässige Wert ist fünf Tage. Wenn Sie einen Wert auf mehr als fünf Tage festlegen, wird er als fünf Tage behandelt, und nach Ablauf der Frist von fünf Tagen werden die Benachrichtigungen erneut gesendet.

> **Note**
>
> Sie können die Konfigurationsdatei auch direkt in der Amazon Managed Service for Prometheus-Konsole bearbeiten, sie muss jedoch weiterhin dem hier angegebenen Format entsprechen. Weitere Informationen zum Hochladen oder Bearbeiten einer Konfigurationsdatei finden Sie unter. Laden Sie Ihre Alert Manager-Konfigurationsdatei auf Amazon Managed Service for Prometheus hoch

In Amazon Managed Service für Prometheus muss Ihre Alert-Manager-Konfigurationsdatei den gesamten Inhalt Ihrer Alert-Manager-Konfiguration in einem `alertmanager_config`-Schlüssel im Stammverzeichnis der YAML-Datei enthalten.

Im Folgenden finden Sie ein einfaches Beispiel für eine Alert-Manager-Konfigurationsdatei:

```
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
```

Der einzige derzeit unterstützte Empfänger ist Amazon Simple Notification Service (Amazon SNS). Wenn Sie andere Arten von Empfängern in der Konfiguration aufgeführt haben, werden diese abgelehnt.

Hier ist ein weiteres Beispiel für eine Alert-Manager-Konfigurationsdatei, die sowohl den Block `template_files` als auch den Block `alertmanager_config` verwendet.

```
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
```

Standardmäßiger Amazon-SNS-Vorlagenblock

Die Amazon-SNS-Standardkonfiguration verwendet die folgende Vorlage, sofern Sie sie nicht explizit überschreiben.

```
{{ 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 }}
```

Leiten Sie Benachrichtigungen mit dem Alert Manager in Amazon Managed Service für Prometheus an einen Alertempfänger weiter

Wenn eine Warnung durch eine Warnungsregel ausgelöst wird, wird sie an den Alert Manager gesendet. Der Alert Manager führt Funktionen wie das Deduplizieren von Alarmen, das Sperren von Warnmeldungen während der Wartung oder das Gruppieren von Alarmen nach Bedarf aus. Anschließend leitet er die Warnung als Nachricht an einen Alarmempfänger weiter. Sie können einen Alarmempfänger einrichten, der die Bediener benachrichtigen, automatische Antworten erhalten oder auf andere Weise auf die Warnungen reagieren kann.

Sie können Amazon Simple Notification Service (Amazon SNS) und PagerDuty als Alertempfänger in Amazon Managed Service für Prometheus konfigurieren. In den folgenden Themen wird beschrieben, wie Sie Ihren Alertempfänger erstellen und konfigurieren.

Themen

- Verwenden Sie Amazon SNS als Alarmempfänger
- PagerDuty Als Alarmempfänger verwenden

Verwenden Sie Amazon SNS als Alarmempfänger

Sie können ein vorhandenes Amazon SNS SNS-Thema als Alertempfänger für Amazon Managed Service for Prometheus verwenden oder ein neues erstellen. Wir empfehlen Ihnen, ein Thema vom Typ Standard zu verwenden, damit Sie Benachrichtigungen von diesem Thema an E-Mail, SMS oder HTTP weiterleiten können.

Um ein neues Amazon-SNS-Thema zu erstellen, das Sie als Alert-Manager-Empfänger verwenden können, folgen Sie den Schritten in Schritt 1: Thema erstellen. Achten Sie darauf, Standard als Thementyp auszuwählen.

Wenn Sie jedes Mal eine E-Mail erhalten möchten, wenn eine Nachricht an dieses Amazon-SNS-Thema gesendet wird, folgen Sie den Schritten in Schritt 2: Abonnement für das Thema erstellen.

Unabhängig davon, ob Sie ein neues oder ein vorhandenes Amazon SNS SNS-Thema verwenden, benötigen Sie den Amazon-Ressourcennamen (ARN) Ihres Amazon SNS-Themas, um die folgenden Aufgaben ausführen zu können.

Themen

- Amazon Managed Service for Prometheus die Erlaubnis erteilen, Warnmeldungen zu Ihrem Amazon SNS SNS-Thema zu senden
- Konfigurieren Sie den Alert Manager so, dass er Nachrichten an Ihr Amazon SNS SNS-Thema sendet
- Konfigurieren Sie den Alert Manager so, dass er Nachrichten als JSON an Amazon SNS sendet
- Amazon SNS so konfigurieren, dass Warnmeldungen an andere Ziele gesendet werden
- Grundlegendes zu den Amazon SNS SNS-Nachrichtenvalidierungsregeln

Amazon Managed Service for Prometheus die Erlaubnis erteilen, Warnmeldungen zu Ihrem Amazon SNS SNS-Thema zu senden

Sie müssen Amazon Managed Service für Prometheus die Erlaubnis erteilen, Nachrichten an Ihr Amazon-SNS-Thema zu senden. In der folgenden Grundsatzerklärung wird diese Genehmigung erteilt. Sie enthält eine `Condition` Erklärung zur Vermeidung eines Sicherheitsproblems, dem sogenannten Confused Deputy Problem. Die `Condition`-Anweisung schränkt den Zugriff auf das Amazon-SNS-Thema so ein, dass nur Vorgänge zugelassen werden, die von diesem bestimmten Konto und Workspace in Amazon Managed Service für Prometheus stammen. Weitere Informationen zum Confused-Deputy-Problem finden Sie Serviceübergreifende Confused-Deputy-Prävention.

Amazon Managed Service für Prometheus die Erlaubnis erteilen, Nachrichten an Ihr Amazon-SNS-Thema zu senden

1. Öffnen Sie die Amazon SNS SNS-Konsole unter https://console.aws.amazon.com/sns/v3/home.
2. Wählen Sie im Navigationsbereich Themen aus.
3. Wählen Sie den Namen des Themas, das Sie mit Amazon Managed Service für Prometheus verwenden.
4. Wählen Sie Bearbeiten aus.
5. Wählen Sie Zugriffsrichtlinie und fügen Sie der vorhandenen Richtlinie die folgende Richtlinienanweisung hinzu.

```
{
    "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"
}
```

[Optional] Wenn für Ihr Amazon SNS SNS-Thema Service Side Encryption (SSE) aktiviert ist, müssen Sie Amazon Managed Service for Prometheus erlauben, Nachrichten an dieses verschlüsselte Thema zu senden, indem Sie die `kms:Decrypt` Berechtigungen `kms:GenerateDataKey*` und zur AWS KMS Schlüsselrichtlinie des Schlüssels hinzufügen, der zur Verschlüsselung des Themas verwendet wird.

Sie könnten der Richtlinie beispielsweise Folgendes hinzufügen:

```
{
  "Statement": [{
    "Effect": "Allow",
    "Principal": {
      "Service": "aps.amazonaws.com"
    },
    "Action": [
      "kms:GenerateDataKey*",
      "kms:Decrypt"
    ],
    "Resource": "*"
  }]
}
```

Weitere Informationen finden Sie unter AWS KMS-Berechtigungen für SNS-Themen.

6. Wählen Sie Änderungen speichern aus.

> **Note**
>
> Standardmäßig erstellt Amazon SNS die Zugriffsrichtlinie mit Bedingung für `AWS:SourceOwner`. Weitere Informationen finden Sie unter SNS-Zugriffrichtlinie.

> **Note**
>
> IAM folgt der Regel Richtlinie zuerst mit den meisten Einschränkungen. Wenn es in Ihrem SNS-Thema einen Richtlinienblock gibt, der restriktiver ist als der dokumentierte Amazon-SNS-Richtlinienblock, wird die Genehmigung für die Themenrichtlinie nicht erteilt. Um Ihre Richtlinie zu bewerten und herauszufinden, was gewährt wurde, finden Sie unter Bewertungslogik für Richtlinien.

SNS-Themenkonfiguration für Opt-in-Regionen

Sie können `aps.amazonaws.com` es verwenden, um ein Amazon SNS SNS-Thema in demselben Arbeitsbereich AWS-Region wie in Ihrem Amazon Managed Service for Prometheus Workspace zu konfigurieren. Um ein SNS-Thema aus einer non-opt-in Region (wie us-east-1) mit einer Opt-in-Region (wie af-south-1) zu verwenden, müssen Sie das Regional Service Principal-Format

verwenden. Ersetzen Sie im Prinzip „Regionaler Dienst“ durch die Region, die non-opt-in Sie *us-east-1* verwenden möchten:. **aps.*us-east-1*.amazonaws.com**

In der folgenden Tabelle sind die Opt-in-Regionen und ihre entsprechenden regionalen Dienstprinzipale aufgeführt:

Opt-in-Regionen und ihre regionalen Diensteanbieter

Name der Region	Region	Leiter des regionalen Dienstes
Afrika (Kapstadt)	af-south-1	af-south-1.aps.amazonaws.com
Asien-Pazifik (Hongkong)	ap-east-1	ap-east-1.aps.amazonaws.com
Asien-Pazifik (Thailand)	ap-southeast-7	ap-southeast-7.aps.amazonaws.com
Europa (Milan)	eu-south-1	eu-south-1.aps.amazonaws.com
Europa (Zürich)	eu-central-2	eu-central-2.aps.amazonaws.com
Naher Osten (VAE)	me-central-1	me-central-1.aps.amazonaws.com
Asien-Pazifik (Malaysia)	ap-southeast-5	ap-southeast-5.aps.amazonaws.com

Informationen zur Aktivierung einer Opt-in-Region finden Sie unter Verwaltung im IAM-Benutzerhandbuch in der. AWS-Regionen Allgemeine Amazon Web Services-Referenz

Stellen Sie bei der Konfiguration Ihres Amazon SNS SNS-Themas für diese Opt-in-Regionen sicher, dass Sie den richtigen regionalen Service Principal verwenden, um die regionsübergreifende Zustellung von Benachrichtigungen zu ermöglichen.

Serviceübergreifende Confused-Deputy-Prävention

Das Confused-Deputy-Problem ist ein Sicherheitsproblem, bei dem eine juristische Stelle, die nicht über die Berechtigung zum Ausführen einer Aktion verfügt, eine privilegiertere juristische Stelle zwingen kann, die Aktion auszuführen. Ein AWS dienstübergreifender Identitätswechsel kann zu einem Problem mit dem verwirrten Stellvertreter führen. Ein dienstübergreifender Identitätswechsel kann auftreten, wenn ein Dienst (der Anruf-Dienst) einen anderen Dienst anruft (den aufgerufenen Dienst). Der aufrufende Service kann manipuliert werden, um seine Berechtigungen zu verwenden, um Aktionen auf die Ressourcen eines anderen Kunden auszuführen, für die er sonst keine Zugriffsberechtigung haben sollte. Um dies zu verhindern, bietet AWS Tools, mit denen Sie Ihre Daten für alle Services mit Serviceprinzipalen schützen können, die Zugriff auf Ressourcen in Ihrem Konto erhalten haben.

Wir empfehlen die Verwendung der globalen Bedingungskontextschlüssel `aws:SourceArn` und `aws:SourceAccount` in ressourcenbasierten Richtlinien, um die Berechtigungen, die Amazon Managed Service für Prometheus Amazon SNS erteilt, auf eine bestimmte Ressource zu beschränken. Wenn Sie beide globalen Bedingungskontextschlüssel verwenden, müssen der `aws:SourceAccount`-Wert und das Konto im `aws:SourceArn`-Wert dieselbe Konto-ID verwenden, wenn sie in derselben Richtlinienanweisung verwendet werden.

Der Wert von `aws:SourceArn` muss der ARN des Workspace in Amazon Managed Service für Prometheus sein.

Der effektivste Weg, um sich vor dem Confused-Deputy-Problem zu schützen, ist die Verwendung des globalen Bedingungskontext-Schlüssels `aws:SourceArn` mit dem vollständigen ARN der Ressource. Wenn Sie den vollständigen ARN der Ressource nicht kennen oder wenn Sie mehrere Ressourcen angeben, verwenden Sie den globalen Bedingungskontext-Schlüssel `aws:SourceArn` mit Platzhaltern (*) für die unbekannten Teile des ARN. Beispiel, `arn:aws:servicename::123456789012:*`.

Das folgende Beispiel in Amazon Managed Service for Prometheus die Erlaubnis erteilen, Warnmeldungen zu Ihrem Amazon SNS SNS-Thema zu senden zeigt, wie Sie die globalen Bedingungskontext-Schlüssel `aws:SourceArn` und `aws:SourceAccount` in Amazon Managed Service für Prometheus verwenden können, um das Problem des Confused Deputys zu vermeiden.

Konfigurieren Sie den Alert Manager so, dass er Nachrichten an Ihr Amazon SNS SNS-Thema sendet

Nachdem Sie ein (neues oder vorhandenes) Amazon SNS SNS-Thema vom Typ Standard erstellt haben, können Sie es Ihrer Alert Manager-Konfiguration als Alert-Empfänger hinzufügen. Der Alert Manager kann Ihre Benachrichtigungen an einen konfigurierten Alertempfänger weiterleiten. Um dies abzuschließen, müssen Sie den Amazon-Ressourcennamen (ARN) Ihres Amazon SNS-Themas kennen.

Weitere Informationen zur Konfiguration des Amazon-SNS-Empfängers finden Sie unter <sns_configs> in der Prometheus-Konfigurationsdokumentation.

Nicht unterstützte Eigenschaften

Amazon Managed Service für Prometheus unterstützt Amazon SNS als Alarmempfänger. Aufgrund von Serviceeinschränkungen werden jedoch nicht alle Eigenschaften des Amazon-SNS-Empfängers unterstützt. Die folgenden Eigenschaften sind in einer Alert-Manager-Konfigurationsdatei von Amazon Managed Service für Prometheus nicht zulässig:

- `api_url:` – Amazon Managed Service für Prometheus legt `api_url` für Sie fest, sodass diese Eigenschaft nicht zulässig ist.
- `Http_config` – Mit dieser Eigenschaft können Sie externe Proxys einrichten. Amazon Managed Service für Prometheus unterstützt diese Funktion derzeit.

Darüber hinaus müssen die SigV4-Einstellungen über die Eigenschaft Region verfügen. Ohne die Eigenschaft Region verfügt Amazon Managed Service für Prometheus nicht über genügend Informationen, um die Autorisierungsanfrage zu stellen.

Um den Alert Manager mit Ihrem Amazon-SNS-Thema als Empfänger zu konfigurieren

1. Wenn Sie eine vorhandene Alert-Manager-Konfigurationsdatei verwenden, öffnen Sie diese in einem Text-Editor.
2. Wenn der `receivers` Block andere aktuelle Empfänger als Amazon SNS enthält, entfernen Sie diese. Sie können mehrere Amazon-SNS-Themen als Empfänger konfigurieren, indem Sie sie innerhalb des Blocks `receivers` in separate Blöcke `sns_config` einordnen.
3. Fügen Sie innerhalb des `receivers` Abschnitts den folgenden YAML-Block hinzu.

```
- name: name_of_receiver
```

```
  sns_configs:
    - sigv4:
        region: AWS-Region
      topic_arn: ARN_of_SNS_topic
      subject: yoursubject
      attributes:
        key: yourkey
        value: yourvalue
```

Wenn `subject` nicht angegeben ist, wird standardmäßig ein Betreff mit der Standardvorlage mit dem Labelnamen und den Werten generiert, was zu einem Wert führen kann, der für SNS zu lang ist. Informationen zum Ändern der Vorlage, die auf den Betreff angewendet wird, finden Sie Konfigurieren Sie den Alert Manager so, dass er Nachrichten als JSON an Amazon SNS sendet in diesem Handbuch.

Jetzt müssen Sie Ihre Alert-Manager-Konfigurationsdatei nach Amazon Managed Service für Prometheus hochladen. Weitere Informationen finden Sie unter Laden Sie Ihre Alert Manager-Konfigurationsdatei auf Amazon Managed Service for Prometheus hoch.

Konfigurieren Sie den Alert Manager so, dass er Nachrichten als JSON an Amazon SNS sendet

Standardmäßig gibt der Amazon Managed Service for Prometheus Alert Manager Nachrichten in einem Nur-Text-Listenformat aus. Dies kann für andere Dienste schwieriger zu analysieren sein. Sie können den Alert Manager so konfigurieren, dass Alerts stattdessen im JSON-Format gesendet werden. JSON kann es einfacher machen, die von Amazon SNS nachgeschalteten Nachrichten in AWS Lambda oder in Webhook-Empfangsendpunkten zu verarbeiten. Anstatt die Standardvorlage zu verwenden, können Sie eine benutzerdefinierte Vorlage für die Ausgabe des Nachrichteninhalts in JSON definieren, was die Analyse in nachgelagerten Funktionen erleichtert.

Um Nachrichten vom Alert Manager an Amazon SNS im JSON-Format auszugeben, aktualisieren Sie Ihre Alert-Manager-Konfiguration so, dass sie den folgenden Code in Ihrem `template_files`-Stammbereich enthält:

```
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**
>
> Diese Vorlage erstellt JSON aus alphanumerischen Daten. Wenn Ihre Daten Sonderzeichen enthalten, kodieren Sie diese, bevor Sie diese Vorlage verwenden.

Um sicherzustellen, dass diese Vorlage in ausgehenden Benachrichtigungen verwendet wird, referenzieren Sie sie in Ihrem `alertmanager_config` Block wie folgt:

```
alertmanager_config: |
  global:
  templates:
    - 'default_template'
```

> **Note**
>
> Diese Vorlage gilt für den gesamten Nachrichtentext als JSON. Diese Vorlage überschreibt den gesamten Nachrichtentext. Sie können den Nachrichtentext nicht überschreiben, wenn Sie diese spezielle Vorlage verwenden möchten. Alle manuell vorgenommenen Überschreibungen haben Vorrang vor der Vorlage.

Weitere Informationen über:

- die Alert-Manager-Konfigurationsdatei, siehe Erstellen Sie in Amazon Managed Service eine Alert Manager-Konfiguration für Prometheus, um Benachrichtigungen zu verwalten und weiterzuleiten.
- das Hochladen Ihrer Konfigurationsdatei, siheLaden Sie Ihre Alert Manager-Konfigurationsdatei auf Amazon Managed Service for Prometheus hoch.

Amazon SNS so konfigurieren, dass Warnmeldungen an andere Ziele gesendet werden

Amazon Managed Service for Prometheus kann nur Warnmeldungen an Amazon Simple Notification Service (Amazon SNS) senden. Um diese Nachrichten an andere Ziele wie E-Mail, Webhook, Slack oder zu senden, müssen Sie Amazon SNS so konfigurieren OpsGenie, dass die Nachrichten an diese Endpunkte weitergeleitet werden.

In den folgenden Abschnitten wird die Konfiguration von Amazon SNS für die Weiterleitung von Benachrichtigungen an andere Ziele beschrieben.

Themen

- Email
- Webhook
- Slack
- OpsGenie

Email

Um ein Amazon-SNS-Thema für die Ausgabe von Nachrichten per E-Mail zu konfigurieren, erstellen Sie ein Abonnement. Wählen Sie in der Amazon-SNS-Konsole den Tab Abonnements, um die Seite mit der Abonnementliste zu öffnen. Wählen Sie Abonnement erstellen und dann E-Mail aus. Amazon SNS sendet eine Bestätigungs-E-Mail an die angegebene E-Mail-Adresse. Nachdem Sie die Bestätigung akzeptiert haben, können Sie Amazon-SNS-Benachrichtigungen als E-Mails zu dem Thema erhalten, das Sie abonniert haben. Weitere Informationen finden Sie unter Tutorial: Abonnieren eines Endpunkts für ein Amazon-SNS-Thema.

Webhook

Um ein Amazon-SNS-Thema für die Ausgabe von Nachrichten an einen Webhook-Endpunkt zu konfigurieren, erstellen Sie ein Abonnement. Wählen Sie in der Amazon-SNS-Konsole den Tab Abonnements, um die Seite mit der Abonnementliste zu öffnen. Wählen Sie Abonnement erstellen und wählen Sie HTTP/HTTPS aus. Nachdem Sie das Abonnement erstellt haben, müssen Sie die Bestätigungsschritte befolgen, um es zu aktivieren. Wenn es aktiv ist, sollte Ihr HTTP-Endpunkt die Amazon-SNS-Benachrichtigungen erhalten. Weitere Informationen finden Sie unter Tutorial: Abonnieren eines Endpunkts für ein Amazon-SNS-Thema. Weitere Informationen zur Nutzung von Slack Webhooks, um Nachrichten an verschiedene Ziele zu veröffentlichen, finden Sie unter Wie verwende ich Webhooks, um Amazon-SNS-Nachrichten in Amazon Chime, Slack oder Microsoft Teams zu veröffentlichen?

Slack

Um ein Amazon-SNS-Thema für die Ausgabe von Nachrichten an Slack zu konfigurieren, haben Sie zwei Möglichkeiten. Du kannst entweder die email-to-channel Integration von Slack integrieren, sodass Slack E-Mail-Nachrichten annehmen und an einen Slack-Channel weiterleiten kann, oder du kannst eine Lambda-Funktion verwenden, um die Amazon SNS-Benachrichtigung an Slack umzuschreiben. Weitere Informationen zum Weiterleiten von E-Mails an Slack-Kanäle findest du unter Bestätigung des SNS-Themenabonnements für Slack Webhook. AWS Weitere Informationen zur Erstellung einer Lambda-Funktion zur Konvertierung von Amazon-SNS-Nachrichten in Slack finden Sie unter So integrieren Sie Amazon Managed Service für Prometheus mit Slack.

OpsGenie

Informationen zur Konfiguration eines Amazon SNS-Themas für die Ausgabe von Nachrichten finden Sie unter Integrieren von Opsgenie mit eingehendem Amazon SNS. OpsGenie

Grundlegendes zu den Amazon SNS SNS-Nachrichtenvalidierungsregeln

Amazon Simple Notification Service (Amazon SNS) setzt voraus, dass Nachrichten bestimmten Standards entsprechen. Nachrichten, die diese Standards nicht erfüllen, werden geändert, wenn sie empfangen werden. Die Warnmeldungen werden vom Amazon SNS SNS-Empfänger auf der Grundlage der folgenden Regeln validiert, gekürzt oder gegebenenfalls geändert:

- Die Nachricht enthält Nicht-UTF-Zeichen.
  - Die Nachricht wird durch Fehler ersetzt — keine gültige UTF-8-kodierte Zeichenfolge.
  - Ein Nachrichtenattribut wird mit dem Schlüssel truncated und dem Wert true hinzugefügt.

  - Es wird ein Nachrichtenattribut mit dem Schlüssel modifiziert und dem Wert Message: Error — keine gültige UTF-8-kodierte Zeichenfolge hinzugefügt.
- Die Nachricht ist leer.
  - Die Nachricht wird durch Fehler ersetzt — Nachricht sollte nicht leer sein.
  - Es wird ein Nachrichtenattribut mit dem Schlüssel modifiziert und dem Wert Message: Error — Message should not be empty hinzugefügt.
- Die Nachricht wurde gekürzt.
  - Die Nachricht wird den gekürzten Inhalt haben.
  - Es wird ein Nachrichtenattribut mit dem Schlüssel truncated und dem Wert true hinzugefügt.
  - Es wird ein Nachrichtenattribut mit dem Schlüssel „modified“ und dem Wert Message: Error — Message has been truncated from ***X*** KB, because it overing the 256 KB size limit, added the 256-KB size limit.
- Der Betreff enthält Steuerzeichen oder Nicht-ASCII-Zeichen.
  - Wenn der Betreff Steuerzeichen oder Nicht-ASCII-Zeichen enthält, ersetzt SNS den Betreff durch Fehler — enthält Kontroll- oder Nicht-ASCII-Zeichen.
  - Entfernen Sie bei Betreffs von SNS-E-Mails Steuerzeichen wie Zeilenumbrüche:. \n
- Der Betreff ist kein ASCII.
  - Der Betreff wird durch Fehler ersetzt — enthält nicht druckbare ASCII-Zeichen.
  - Es wird ein Nachrichtenattribut mit dem Schlüssel geändert und dem Wert Betreff: Fehler — enthält nicht druckbare ASCII-Zeichen hinzugefügt.
- Der Betreff wurde gekürzt.
  - Im Betreff wird der Inhalt gekürzt.
  - Es wird ein Nachrichtenattribut mit dem Schlüssel geändert und dem Wert Betreff: Fehler - Der Betreff wurde um alle Zeichen gekürzt, da der Betreff die Größenbeschränkung von 100 ***X*** Zeichen überschreitet.
- Das Nachrichtenattribut hat einen ungültigen Schlüssel/Wert.
  - Ungültiges Nachrichtenattribut wird entfernt.
  - Es wird ein Nachrichtenattribut mit dem Schlüssel „geändert“ und dem Wert „FehlerMessageAttribute“ hinzugefügt, falls die Nachrichtenattribute aufgrund ***X*** des Fehlers ungültig MessageAttributeKey oder MessageAttributeValue entfernt wurden.
- Das Nachrichtenattribut wurde gekürzt.
  - Zusätzliche Nachrichtenattribute werden entfernt.

- Es wird ein Nachrichtenattribut mit dem Schlüssel „geändert“ und dem Wert „FehlerMessageAttribute“ hinzugefügt, wenn die Nachrichtenattribute entfernt wurden, weil es die Größenbeschränkung ***X*** von 256 KB überschreitet.

PagerDuty Als Alarmempfänger verwenden

Sie können Amazon Managed Service for Prometheus so konfigurieren, dass Benachrichtigungen direkt an gesendet werden. PagerDuty Für diese Integration müssen Sie Ihren PagerDuty Integrationsschlüssel in speichern AWS Secrets Manager und Amazon Managed Service for Prometheus die Erlaubnis erteilen, das Geheimnis zu lesen.

PagerDuty Die Integration ermöglicht automatisierte Workflows zur Reaktion auf Vorfälle und stellt sicher, dass kritische Benachrichtigungen die richtigen Teammitglieder zur richtigen Zeit erreichen. Wenn Sie es PagerDuty als Alarmempfänger verwenden, können Sie die Vorteile der PagerDuty Eskalationsrichtlinien, der Bereitschaftsplanung und der Incident-Management-Funktionen nutzen, um sicherzustellen, dass Alarme schnell bestätigt und behoben werden. Diese Integration ist besonders nützlich für Produktionsumgebungen, in denen eine schnelle Reaktion auf Systemprobleme für die Aufrechterhaltung der Serviceverfügbarkeit und die Erfüllung der SLA-Anforderungen unerlässlich ist. Weitere Informationen finden Sie in der PagerDuty Knowledge Base auf der PagerDuty Website.

PagerDuty Konfigurationsoptionen

Option	Description	Erforderlich
`routing_key`	Der PagerDuty Routingsc hlüssel für eine Integration in einem Dienst. Sie müssen dies als Secrets Manager ARN angeben	Ja
`service_key`	Der PagerDuty Dienstsch lüssel für eine Integration in einen Dienst. Sie müssen dies als Secrets Manager ARN angeben	Ja (für Events API v1)

Option	Description	Erforderlich
`client`	Die Client-ID des Notifiers	Nein
`client_url`	Ein Backlink zum Absender der Benachrichtigung	Nein
`description`	Beschreibung des Vorfalls	Nein
`details`	Eine Reihe von beliebigen key/value Paaren, die weitere Informationen über den Vorfall liefern	Nein
`severity`	Schwere des Vorfalls	Nein
`class`	Die Klasse oder Art des Ereignisses	Nein
`component`	Komponente der Quellmasc hine, die für das Ereignis verantwortlich ist	Nein
`group`	Logische Gruppierung von Komponenten	Nein
`source`	Der eindeutige Standort des betroffenen Systems	Nein

> **Note**
>
> Die `http_config` Optionen `urlservice_key_file`,`routing_key_file`, und werden nicht unterstützt.

In den folgenden Themen wird beschrieben, wie Sie in Amazon Managed Service for Prometheus PagerDuty als Alert-Empfänger konfigurieren.

Themen

- Konfiguration AWS Secrets Manager und Berechtigungen
- Konfigurieren Sie den Alert Manager so, dass Benachrichtigungen gesendet werden PagerDuty

Konfiguration AWS Secrets Manager und Berechtigungen

Bevor Sie Benachrichtigungen an senden können PagerDuty, müssen Sie Ihren PagerDuty Integrationsschlüssel sicher speichern und die erforderlichen Berechtigungen konfigurieren. Bei diesem Vorgang wird ein Secret in erstellt AWS Secrets Manager, es mit einem vom Kunden verwalteten AWS Key Management Service (AWS KMS) Schlüssel verschlüsselt und Amazon Managed Service for Prometheus die erforderlichen Berechtigungen für den Zugriff sowohl auf das Geheimnis als auch auf seinen Verschlüsselungsschlüssel gewährt. Die folgenden Verfahren führen Sie durch die einzelnen Schritte dieses Konfigurationsprozesses.

Um ein Geheimnis in Secrets Manager zu erstellen für PagerDuty

Um ihn PagerDuty als Alertempfänger zu verwenden, müssen Sie Ihren PagerDuty Integrationsschlüssel in Secrets Manager speichern. Dazu gehen Sie wie folgt vor:

1. Öffnen Sie die Secrets Manager-Konsole.
2. Wählen Sie Store a new secret (Ein neues Secret speichern).
3. Als Secret-Typ wählen Sie Anderer Secret-Typ aus.
4. Geben Sie für Schlüssel/Wert-Paare Ihren PagerDuty Integrationsschlüssel als geheimen Wert ein. Dies ist entweder der Routingschlüssel oder der Serviceschlüssel aus Ihrer PagerDuty Integration.
5. Wählen Sie Weiter aus.
6. Geben Sie einen Namen und eine Beschreibung für Ihr Geheimnis ein und wählen Sie dann Weiter.
7. Konfigurieren Sie bei Bedarf die Rotationseinstellungen und wählen Sie dann Weiter.
8. Überprüfe deine Einstellungen und wähle „Speichern“.
9. Notieren Sie sich nach der Erstellung des Geheimnisses seinen ARN. Sie benötigen dies, wenn Sie den Alert Manager konfigurieren.

Um Ihr Geheimnis mit einem vom Kunden AWS KMS verwalteten Schlüssel zu verschlüsseln

Sie müssen Amazon Managed Service for Prometheus die Erlaubnis erteilen, auf Ihr Geheimnis und dessen Verschlüsselungsschlüssel zuzugreifen:

1. Secret Resource Policy: Öffnen Sie Ihr Secret in der Secrets Manager Manager-Konsole.

   a. Wählen Sie Ressourcenberechtigungen.

   b. Wählen Sie Berechtigungen bearbeiten aus.

   c. Fügen Sie die folgende Richtlinienerklärung hinzu. Ersetzen Sie in der Erklärung die *highlighted values* durch Ihre spezifischen Werte.

```
{
  "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. Wählen Sie Speichern.

2. KMS-Schlüsselrichtlinie: Öffnen Sie Ihren AWS KMS Schlüssel in der AWS KMS Konsole.

   a. Wählen Sie Schlüsselrichtlinie aus.

   b. Wählen Sie Bearbeiten aus.

   c. Fügen Sie die folgende Richtlinienerklärung hinzu. Ersetzen Sie in der Erklärung die *highlighted values* durch Ihre spezifischen Werte.

```
{
  "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. Wählen Sie Speichern.

Nächste Schritte — Fahren Sie mit dem nächsten Thema fort,Konfigurieren Sie den Alert Manager so, dass Benachrichtigungen gesendet werden PagerDuty.

Konfigurieren Sie den Alert Manager so, dass Benachrichtigungen gesendet werden PagerDuty

Um den Alert Manager für das Senden von Benachrichtigungen zu konfigurieren PagerDuty, müssen Sie Ihre Alert Manager-Definition aktualisieren. Sie können dies mit dem AWS-Managementkonsole AWS CLI, oder tun AWS SDKs.

Example Konfiguration des Alert Managers

Im Folgenden finden Sie ein Beispiel für eine Alert Manager-Konfiguration, an die Warnmeldungen gesendet PagerDuty werden. Ersetzen Sie im Beispiel die *highlighted values* durch Ihre spezifischen Werte.

```
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

Im Folgenden finden AWS CLI Sie einen Befehl, mit dem Sie Ihre Alert Manager-Definition aktualisieren können. Ersetzen Sie im Beispiel die *highlighted values* durch Ihre spezifischen Werte.

```
aws amp put-alert-manager-definition \
  --workspace-id WORKSPACE_ID \
  --data file://alertmanager-config.yaml
```

Fehlerbehebung bei der PagerDuty Integration

Wenn keine Benachrichtigungen gesendet werden PagerDuty, überprüfen Sie die folgenden Punkte:

- Stellen Sie sicher, dass Ihr Geheimnis existiert und den richtigen PagerDuty Integrationsschlüssel enthält.
- Vergewissern Sie sich, dass Ihr Geheimnis mit einem vom Kunden verwalteten KMS-Schlüssel verschlüsselt ist.
- Stellen Sie sicher, dass die Ressourcenrichtlinien sowohl für den geheimen als auch für den KMS-Schlüssel Amazon Managed Service for Prometheus die erforderlichen Berechtigungen gewähren.
- Vergewissern Sie sich, dass der ARN in Ihrer Alert Manager-Konfiguration korrekt auf Ihr Geheimnis verweist.
- Stellen Sie sicher, dass Ihr PagerDuty Integrationsschlüssel gültig und in Ihrem PagerDuty Konto aktiv ist.

Amazon Managed Service for Prometheus unterstützt Amazon CloudWatch Logs und die folgenden CloudWatch Metriken, um bei der Fehlerbehebung zu helfen. Weitere Informationen erhalten

Sie unter Überwachen Sie Amazon Managed Service für Prometheus-Ereignisse mit Protokollen CloudWatch und Verwenden Sie CloudWatch Metriken, um die Ressourcen von Amazon Managed Service for Prometheus zu überwachen.

- `SecretFetchFailure`
- `AlertManagerNotificationsThrottledByIntegration`
- `AlertManagerNotificationsFailedByIntegration`

Laden Sie Ihre Alert Manager-Konfigurationsdatei auf Amazon Managed Service for Prometheus hoch

Sobald Sie wissen, was Sie in Ihrer Alert Manager-Konfigurationsdatei haben möchten, können Sie sie in der Konsole erstellen und bearbeiten, oder Sie können eine bestehende Datei mit der Amazon Managed Service for Prometheus-Konsole hochladen oder. AWS CLI

> **Note**
>
> Wenn Sie einen Amazon EKS-Cluster ausführen, können Sie mit AWS Controllers for Kubernetes auch eine Alert Manager-Konfigurationsdatei hochladen.

So verwenden Sie die Amazon Managed Service for Prometheus-Konsole, um Ihre Alert Manager-Konfiguration zu bearbeiten oder zu ersetzen

1. Öffnen Sie die Amazon Managed Service for Prometheus-Konsole unter. https://console.aws.amazon.com/prometheus/
2. Wählen Sie in der oberen linken Ecke der Seite das Menüsymbol und dann Alle Workspaces aus.
3. Wählen Sie die Workspace-ID des Workspace und dann den Tab Alert Manager aus.
4. Wenn der Workspace noch keine Alert-Manager-Definition hat, wählen Sie Definition hinzufügen.

   > **Note**
   >
   > Wenn der Workspace über eine Alert Manager-Definition verfügt, die Sie ersetzen möchten, wählen Sie stattdessen Ändern.

5. Wählen Sie Datei auswählen, wählen Sie die Alert-Manager-Definitionsdatei aus und klicken Sie auf Weiter.

> **Note**
>
> Alternativ können Sie eine neue Datei erstellen und sie direkt in der Konsole bearbeiten, indem Sie die Option Definition erstellen wählen. Dadurch wird eine Beispiel-Standardkonfiguration erstellt, die Sie vor dem Hochladen bearbeiten.

Um den zu verwenden AWS CLI , um eine Alert Manager-Konfiguration zum ersten Mal in einen Workspace hochzuladen

1. Kodiert den Inhalt Ihrer Alert-Manager-Datei mit Base64. Sie können für Linux den folgenden Befehl verwenden:

```
base64 input-file output-file
```

   Sie können für macOS den folgenden Befehl verwenden:

```
openssl base64 input-file output-file
```

2. Geben Sie folgenden Befehle ein, um die Datei hochzuladen.

   Geben Sie in AWS CLI Version 2 Folgendes ein:

```
aws amp create-alert-manager-definition --data file://path_to_base_64_output_file
 --workspace-id my-workspace-id --region region
```

   Geben Sie in AWS CLI Version 1 Folgendes ein:

```
aws amp create-alert-manager-definition --data fileb://path_to_base_64_output_file
 --workspace-id my-workspace-id --region region
```

3. Es dauert einige Sekunden, bis Ihre Alert-Manager-Konfiguration aktiv wird. Geben Sie zur Prüfung des Status den folgenden Befehl ein:

```
aws amp describe-alert-manager-definition --workspace-id workspace_id --
region region
```

Wenn `status` `ACTIVE` ist, ist Ihre neue Alert-Manager-Definition wirksam geworden.

AWS CLI Um die Alert Manager-Konfiguration eines Workspace durch eine neue zu ersetzen

1. Kodiert den Inhalt Ihrer Alert-Manager-Datei mit Base64. Sie können für Linux den folgenden Befehl verwenden:

   ```
   base64 input-file output-file
   ```

   Sie können für macOS den folgenden Befehl verwenden:

   ```
   openssl base64 input-file output-file
   ```

2. Geben Sie folgenden Befehle ein, um die Datei hochzuladen.

   Geben Sie in AWS CLI Version 2 Folgendes ein:

   ```
   aws amp put-alert-manager-definition --data file://path_to_base_64_output_file --
   workspace-id my-workspace-id --region region
   ```

   Geben Sie in AWS CLI Version 1 Folgendes ein:

   ```
   aws amp put-alert-manager-definition --data file://path_to_base_64_output_file --
   workspace-id my-workspace-id --region region
   ```

3. Es dauert einige Sekunden, bis Ihre neue Alert-Manager-Konfiguration aktiv wird. Geben Sie zur Prüfung des Status den folgenden Befehl ein:

   ```
   aws amp describe-alert-manager-definition --workspace-id workspace_id --
   region region
   ```

   Wenn `status` `ACTIVE` ist, ist Ihre neue Alert-Manager-Definition wirksam geworden. Bis zu diesem Zeitpunkt ist Ihre vorherige Alert-Manager-Konfiguration noch aktiv.

Integrieren Sie Benachrichtigungen in Amazon Managed Grafana oder Open-Source-Grafana

Alarmregeln, die Sie in Alertmanager innerhalb von Amazon Managed Service für Prometheus erstellt haben, können in Amazon Managed Grafana und Grafana weitergeleitet und angezeigt werden, wodurch Ihre Alarmregeln und Alarme in einer einzigen Umgebung vereinheitlicht werden. In Amazon Managed Grafana können Sie Ihre Alarmregeln und die generierten Alarme anzeigen.

Voraussetzungen

Bevor Sie mit der Integration von Amazon Managed Service für Prometheus in Amazon Managed Grafana beginnen, müssen Sie die folgenden Voraussetzungen erfüllt haben:

- Sie benötigen bereits vorhandene AWS-Konto und IAM-Anmeldeinformationen, um Amazon Managed Service for Prometheus- und IAM-Rollen programmgesteuert zu erstellen.

  Weitere Informationen zum Erstellen von und IAM-Anmeldeinformationen finden Sie unter. AWS-Konto Einrichten AWS
- Sie müssen über einen Workspace in Amazon Managed Service für Prometheus verfügen und Daten in diesem erfassen. Informationen zum Einrichten eines neuen Workspace finden Sie unter Erstellen eines Workspace in Amazon Managed Service für Prometheus. Sie sollten auch mit den Prometheus-Konzepten wie Alertmanager und Ruler vertraut sein. Weitere Informationen zu diesen Themen finden Sie in der Prometheus-Dokumentation.
- Sie haben bereits eine Alertmanager-Konfiguration und eine Regeldatei in Amazon Managed Service für Prometheus konfiguriert. Weitere Informationen über Alertmanager in Amazon Managed Service für Prometheus finden Sie in Verwaltung und Weiterleitung von Benachrichtigungen in Amazon Managed Service für Prometheus mit Alert Manager. Weitere Informationen zu Regeln finden Sie unter Verwenden von Regeln zum Ändern oder Überwachen von Metriken, sobald sie empfangen werden.
- Sie müssen entweder Amazon Managed Grafana eingerichtet oder die Open-Source-Version von Grafana ausgeführt haben.
  - Wenn Sie Amazon Managed Grafana verwenden, müssen Sie Grafanaalarme verwenden. Weitere Informationen finden Sie unter Migration älterer Dashboard-Alarme zu Grafana-Alarme.
  - Wenn Sie die Open-Source-Version von Grafana verwenden, müssen Sie Version 9.1 oder höher ausführen.

> **Note**
>
> Sie können frühere Versionen von Grafana verwenden, müssen jedoch die Funktion Unified Alerting (Grafana-Alarme) aktivieren und möglicherweise einen SigV4-Proxy einrichten, um Aufrufe von Grafana an Amazon Managed Service für Prometheus zu tätigen. Weitere Informationen finden Sie unter Richten Sie Grafana Open Source oder Grafana Enterprise für die Verwendung mit Amazon Managed Service für Prometheus ein.

- Amazon Managed Grafana muss die folgenden Berechtigungen für Ihre Prometheus-Ressourcen besitzen. Sie müssen sie entweder zu den vom Service verwalteten oder den vom Kunden verwalteten Richtlinien hinzufügen, die unter https://docs.aws.amazon.com/grafana/latest/userguide/AMG-manage-permissions.html beschrieben sind.
  - `aps:ListRules`
  - `aps:ListAlertManagerSilences`
  - `aps:ListAlertManagerAlerts`
  - `aps:GetAlertManagerStatus`
  - `aps:ListAlertManagerAlertGroups`
  - `aps:PutAlertManagerSilences`
  - `aps:DeleteAlertManagerSilence`

Einrichten von Amazon Managed Grafana

Wenn Sie in Ihrer Instance von Amazon Managed Service für Prometheus bereits Regeln und Alarme eingerichtet haben, erfolgt die Konfiguration zur Verwendung von Amazon Managed Grafana als Dashboard für diese Alarme vollständig in Amazon Managed Grafana.

So konfigurieren Sie Amazon Managed Grafana als Ihr Alarm-Dashboard

1. Öffnen Sie die Grafana-Konsole für Ihren Workspace.
2. Wählen Sie unter Konfigurationen die Option Datenquellen aus.
3. Erstellen oder öffnen Sie Ihre Prometheus-Datenquelle. Wenn Sie noch keine Prometheus-Datenquelle eingerichtet haben, finden Sie weitere Informationen unter Schritt 2: Fügen Sie die Prometheus-Datenquelle in Grafana hinzu.

4. Wählen Sie in der Prometheus-Datenquelle die Option Alarme über die Alertmanager-Benutzeroberfläche verwalten aus.
5. Kehren Sie zur Datenquellen-Oberfläche zurück.
6. Erstellen Sie eine neue Alertmanager-Datenquelle.
7. Fügen Sie auf der Konfigurationsseite für die Alertmanager-Datenquelle die folgenden Einstellungen hinzu:

   - Stellen Sie Implementierung auf `Prometheus` ein.
   - Verwenden Sie für die URL-Einstellung die URL für Ihren Prometheus-Workspace, entfernen Sie alles, was hinter der Workspace-ID steht, und fügen Sie `/alertmanager` am Ende an. Ersetzen Sie im folgenden Beispiel die *`variables`* durch Ihre eigenen (kontospezifischen) Informationen:

     ```
     https://aps-workspaces.US East (N. Virginia).amazonaws.com/workspaces/ws-example-1234-5678-abcd-xyz00000001/alertmanager.
     ```

   - Aktivieren Sie unter Auth die Option SigV4Auth. Dadurch wird Grafana angewiesen, die AWS Authentifizierung für die Anfragen zu verwenden.
   - Geben Sie unter SIGV4Auth-Details für Standardregion die Region Ihrer Prometheus-Instance an, z. B. `us-east-1`.
   - Stellen Sie die Standardoption auf `true` ein.
8. Klicken Sie auf Speichern und Testen.
9. Ihre Alarme für Amazon Managed Service für Prometheus sollten jetzt so konfiguriert sein, dass sie mit Ihrer Grafana-Instance funktionieren. Vergewissern Sie sich, dass Sie alle Alarmregeln, Alarmgruppen (einschließlich aktiver Alarme) und Stummschaltungen aus Ihrer Instance von Amazon Managed Service für Prometheus auf der Grafana-Alarm-Seite sehen können.

Beheben Sie Fehler im Alert Manager mithilfe von CloudWatch Protokollen

Mit Überwachen Sie Amazon Managed Service für Prometheus-Ereignisse mit Protokollen CloudWatch können Sie Probleme im Zusammenhang mit Alert Manager und Ruler beheben. Dieser Abschnitt enthält Themen zur Fehlerbehebung im Zusammenhang mit Alert Manager.

Themen

- Warnung vor aktiven Alerts
- Warnung zur Größe der Alert-Aggregationsgruppe
- Die Größe der Warnmeldungen ist zu groß, Warnung
- Warnung vor leerem Inhalt
- Ungültige key/value Warnung
- Warnung vor Nachrichtenlimit
- Kein ressourcenbasierter Richtlinienfehler
- Nicht-ASCII-Warnung
- Nicht berechtigt, KMS anzurufen
- Vorlagenfehler

Warnung vor aktiven Alerts

Wenn das Protokoll die folgende Warnung enthält

```
{
    "workspaceId": "ws-efdc5b42-b051-11ec-b123-4567ac120002",
    "message": {
        "log": "too many alerts, limit: 1000",
        "level": "WARN"
    },
    "component": "alertmanager"
}
```

Dies bedeutet, dass das Kontingent für aktive Alerts des Alert Managers überschritten wurde.

Maßnahme

Anfordern einer Kontingenterhöhung. Melden Sie sich bei der an AWS-Managementkonsole und öffnen Sie die Service Quotas Quotas-Konsole unter https://console.aws.amazon.com/servicequotas/.

Warnung zur Größe der Alert-Aggregationsgruppe

Wenn das Protokoll die folgende Warnung enthält

```
{
```

```
    "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"
}
```

Dies bedeutet, dass das Größenkontingent für die Alert Manager-Alert-Aggregationsgruppe überschritten wurde.

Maßnahme

Reduzieren Sie die Größe der Alert-Aggregationsgruppe mithilfe des `group_by` Parameters. Weitere Informationen finden Sie in der Prometheus-Dokumentation unter Route-bezogene Einstellungen.

Sie können auch eine Kontingenterhöhung beantragen. Melden Sie sich bei der an AWS-Managementkonsole und öffnen Sie die Service Quotas Quotas-Konsole unter https://console.aws.amazon.com/servicequotas/.

Die Größe der Warnmeldungen ist zu groß, Warnung

Wenn das Protokoll die folgende Warnung enthält

```
{
    "workspaceId": "ws-efdc5b42-b051-11ec-b123-4567ac120002",
    "message": {
        "log": "alerts too big, total size limit: 20000000 bytes",
        "level": "WARN"
    },
    "component": "alertmanager"
}
```

Das bedeutet, dass das Größenkontingent für Alert Manager-Alerts pro Arbeitsbereich überschritten wurde.

Maßnahme

Entfernen Sie unnötige Anmerkungen und Beschriftungen, um die Größe der Warnmeldungen zu reduzieren.

Warnung vor leerem Inhalt

Wenn das Protokoll die folgende Warnung enthält

```
{
   "workspaceId": "ws-abcd1234-ef56-78ab-cd90-1234abcd0000",
   "message": {
       "log": "Message has been modified because the content was empty."
       "level": "WARN"
   },
   "component": "alertmanager"
}
```

Das bedeutet, dass die Alert-Manager-Vorlage die ausgehende Warnung in eine leere Nachricht aufgelöst hat.

Maßnahme

Validieren Sie Ihre Alert-Manager-Vorlage und stellen Sie sicher, dass Sie über eine gültige Vorlage für alle Empfängerpfade verfügen.

Ungültige **`key/value`** Warnung

Wenn das Protokoll die folgende Warnung enthält

```
{
   "workspaceId": "ws-abcd1234-ef56-78ab-cd90-1234abcd0000",
   "message": {
       "log": "MessageAttributes has been removed because of invalid key/value,
 numberOfRemovedAttributes=1"
       "level": "WARN"
   },
   "component": "alertmanager"
}
```

Das bedeutet, dass einige Nachrichtenattribute entfernt wurden, weil keys/values sie ungültig sind.

Maßnahme

Prüfen Sie die Vorlagen erneut, die Sie zum Füllen der Nachrichtenattribute verwenden, und stellen Sie sicher, dass sie in ein gültiges SNS-Nachrichtenattribut aufgelöst werden. Weitere Informationen zum Überprüfen einer Nachricht an ein Amazon-SNS-Thema finden Sie unter SNS-Thema validieren

Warnung vor Nachrichtenlimit

Wenn das Protokoll die folgende Warnung enthält

```
{
   "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"
}
```

Das bedeutet, dass ein Teil der Nachrichtengröße zu groß ist.

Maßnahme

Sehen Sie sich die Nachrichtenvorlage für den Alert-Empfänger an und überarbeiten Sie sie so, dass sie innerhalb der Größenbeschränkung liegt.

Kein ressourcenbasierter Richtlinienfehler

Wenn das Protokoll den folgenden Fehler enthält

```
{
   "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"
}
```

Das bedeutet, dass Amazon Managed Service für Prometheus nicht über die erforderlichen Rechte verfügt, um die Warnung an das angegebene SNS-Thema zu senden.

Maßnahme

Stellen Sie sicher, dass die Zugriffsrichtlinie für Ihr Amazon-SNS-Thema Amazon Managed Service for Prometheus die Möglichkeit einräumt, SNS-Nachrichten an das Thema zu senden. Erstellen Sie eine SNS-Zugriffsrichtlinie, die dem Service `aps.amazonaws.com` (Amazon Managed Service for Prometheus) Zugriff auf Ihr Amazon SNS SNS-Thema gewährt. Weitere Informationen zu SNS-Zugriffsrichtlinien finden Sie unter Using the Access Policy Language und Beispielfälle für die Amazon SNS SNS-Zugriffskontrolle im Amazon Simple Notification Service Developer Guide.

Nicht-ASCII-Warnung

Wenn das Protokoll die folgende Warnung enthält

```
{
   "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"
}
```

Das bedeutet, dass der Betreff Nicht-ASCII-Zeichen enthält.

Maßnahme

Entfernen Sie Verweise im Betrefffeld Ihrer Vorlage auf den Beschriftungen, die möglicherweise Nicht-ASCII-Zeichen enthalten.

Nicht berechtigt, KMS anzurufen

Wenn das Protokoll den folgenden AWS KMS Fehler enthält

```
{
   "workspaceId": "ws-abcd1234-ef56-78ab-cd90-1234abcd0000",
   "message": {
       "log": "Notify for alerts failed, AMP is not authorized to call KMS",
       "level": "ERROR"
   },
   "component": "alertmanager"
}
```

Maßnahme

Stellen Sie sicher, dass die Schlüsselrichtlinie des Schlüssels, der zur Verschlüsselung des Amazon SNS SNS-Themas verwendet wird, es dem Amazon Managed Service for Prometheus Service Service for Prometheus Service Service Service for Prometheus ermöglicht`aps.amazonaws.com`, die folgenden Aktionen durchzuführen:, und. `kms:GenerateDataKey*` `kms:Decrypt` Weitere Informationen finden Sie unter AWS KMS-Berechtigungen für SNS-Themen.

Vorlagenfehler

Wenn das Protokoll den folgenden Fehler enthält

```
             {
   "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"
}
```

Dies bedeutet, dass ein Fehler in einer Vorlage vorliegt, die in der AlertManager Definition verwendet wird. Der Fehlereintrag enthält Anweisungen zum Empfänger, zur Position in den sns_configs und zu der Eigenschaft, die Fehler enthält.

Maßnahme

Überprüfen Sie Ihre Alert Manager-Definition. Stellen Sie sicher, dass die Syntax korrekt ist und dass Sie auf vorhandene Vorlagenvariablen und Funktionen verweisen. Weitere Informationen finden Sie in der Notification Template Reference in der Open-Source-Dokumentation von Prometheus.

Protokollierung und Überwachung von Amazon Managed Service für Prometheus-Workspaces

Amazon Managed Service for Prometheus verwendet Amazon CloudWatch , um Daten über seinen Betrieb bereitzustellen. Sie können CloudWatch Metriken verwenden, um mehr über die Ressourcennutzung und Anfragen an Ihre Amazon Managed Service for Prometheus-Workspaces zu erfahren. Sie können die CloudWatch Logs-Unterstützung aktivieren, um Protokolle für Ereignisse in Ihren Workspaces abzurufen.

In den folgenden Themen wird die Verwendung CloudWatch ausführlicher beschrieben.

Verwenden Sie CloudWatch Metriken, um die Ressourcen von Amazon Managed Service for Prometheus zu überwachen

Amazon Managed Service for Prometheus verkauft Nutzungsmetriken an. CloudWatch Diese Metriken geben Aufschluss über Ihre Workspace-Nutzung. Die verkauften Metriken finden Sie in den Namespaces und in. `AWS/Usage` `AWS/Prometheus` CloudWatch Diese Metriken sind kostenlos erhältlich. CloudWatch Weitere Informationen zu Nutzungsmetriken finden Sie unter CloudWatch - Nutzungsmetriken.

CloudWatch Name der Metrik	Ressourcennname	CloudWatch Namespace	Description
ResourceCount*	CreateAlertManagerAlertsTPS	`AWS/Usage`	Die maximale Anzahl von `CreateAlertManagerAlerts` API-Vorgängen pro Sekunde pro Workspace
ResourceCount*	DeleteAlertManagerSilencesTPS	`AWS/Usage`	Die maximale Anzahl von `DeleteAlertManagerSilences` API-Vorgängen pro Sekunde pro Workspace

CloudWatch Name der Metrik	Ressourcenname	CloudWatch Namespace	Description
ResourceCount*	GetAlertManagerSilenceTPS	AWS/Usage	Die maximale Anzahl von GetAlertManagerSilence API-Vorgängen pro Sekunde pro Workspace
ResourceCount*	GetAlertManagerStatusTPS	AWS/Usage	Die maximale Anzahl von GetAlertManagerStatus API-Vorgängen pro Sekunde pro Workspace
ResourceCount*	GetLabelsTPS	AWS/Usage	Die maximale Anzahl von GetLabels API-Vorgängen pro Sekunde pro Workspace
ResourceCount*	GetMetricMetadataTPS	AWS/Usage	Die maximale Anzahl von GetMetricMetadata API-Vorgängen pro Sekunde pro Workspace
ResourceCount*	GetSeriesTPS	AWS/Usage	Die maximale Anzahl von GetSeries API-Vorgängen pro Sekunde pro Workspace
ResourceCount	InhibitionRulesInAlertManagerDefinition	AWS/Usage	Die maximale Anzahl von Unterdrückungsregeln in der Alert-Manager-Definitionsdatei.
ResourceCount*	ListAlertManagerAlertGroupInfosTPS	AWS/Usage	Die maximale Anzahl von ListAlertManagerAlertGroupInfos API-Vorgängen pro Sekunde pro Workspace

CloudWatch Name der Metrik	Ressource nname	CloudWatch Namespace	Description
ResourceCount*	ListAlert ManagerAl ertGroupsTPS	`AWS/Usage`	Die maximale Anzahl von `ListAlertManagerAl ertGroups` API-Vorgä ngen pro Sekunde pro Workspace
ResourceCount*	ListAlert ManagerAl ertsTPS	`AWS/Usage`	Die maximale Anzahl von `ListAlertManagerAl erts` API-Vorgängen pro Sekunde pro Workspace
ResourceCount*	ListAlert ManagerRe ceiversTPS	`AWS/Usage`	Die maximale Anzahl von `ListAlertManagerRe ceivers` API-Vorgä ngen pro Sekunde pro Workspace
ResourceCount*	ListAlert ManagerSi lencesTPS	`AWS/Usage`	Die maximale Anzahl von `ListAlertManagerSi lences` API-Vorgä ngen pro Sekunde pro Workspace
ResourceCount*	ListAlertsTPS	`AWS/Usage`	Die maximale Anzahl von `ListAlerts` API-Vorgä ngen pro Sekunde pro Workspace
ResourceCount*	ListRulesTPS	`AWS/Usage`	Die maximale Anzahl von `ListRules` API-Vorgä ngen pro Sekunde pro Workspace

CloudWatch Name der Metrik	Ressourcennname	CloudWatch Namespace	Description
ResourceCount*	PutAlertManagerSilencesTPS	AWS/Usage	Die maximale Anzahl von PutAlertManagerSilences API-Vorgängen pro Sekunde pro Workspace
ResourceCount	HAReplicaGroupCount	AWS/Usage	Anzahl der Replikatgruppen mit hoher Verfügbarkeit
ResourceCount*	QueryMetricsTPS	AWS/Usage	Abfragevorgänge pro Sekunde
ResourceCount*	RemoteWriteTPS	AWS/Usage	Fernschreibvorgänge pro Sekunde
ResourceCount	ActiveAlerts	AWS/Usage	Anzahl der aktiven Alarme pro Workspace Einheiten: Anzahl Gültige Statistiken: Durchschnitt, Minimum, Maximum
ResourceCount	ActiveSeries	AWS/Usage	Anzahl der aktiven Serien pro Workspace Einheiten: Anzahl Gültige Statistiken: Durchschnitt, Minimum, Maximum

CloudWatch Name der Metrik	Ressourcennname	CloudWatch Namespace	Description
ResourceCount	AlertAggregationGroupSize	AWS/Usage	Die maximale Größe einer Alarmaggregationsgruppe in der Alert-Manager-Definitionsdatei. Jede Kombination von Labelwerten `group_by` würde eine Aggregationsgruppe bilden.
ResourceCount	AlertManagerDefinitionSizeBytes	AWS/Usage	Die maximale Größe einer Warnungs-Manager-Definitionsdatei in Byte.
ResourceCount	AllSilences	AWS/Usage	Maximale Anzahl von Stillständen, einschließlich abgelaufener, aktiver und ausstehender Stille, pro Arbeitsbereich.
ResourceCount	IngestionRate	AWS/Usage	Beispielerfassungsrate Einheiten: Anzahl pro Sekunde Gültige Statistiken: Durchschnitt, Minimum, Maximum
ResourceCount	RuleEvaluationInterval	AWS/Usage	Das Mindestintervall für die Regelauswertung
ResourceCount	RuleGroupNamespaceDefinitionSizeBytes	AWS/Usage	Die maximale Größe einer Regelgruppen-Namespace-Definitionsdatei in Byte.

CloudWatch Name der Metrik	Ressource nname	CloudWatch Namespace	Description
ResourceCount	Templates InAlertMa nagerDefinition	AWS/Usage	Die maximale Anzahl der Vorlagen in der Alert-Man ager-Definitionsdatei.
ResourceCount	WorkspaceCount	AWS/Usage	Die maximale Anzahl von Workspaces pro Region, pro Konto.
ResourceCount	SizeOfAlerts	AWS/Usage	Gesamtgröße aller Benachrichtigungen im Workspace, in Byte Einheiten: Byte Gültige Statistiken: Durchschnitt, Minimum, Maximum
ResourceCount	Suppresse dAlerts	AWS/Usage	Anzahl der Alarme im unterdrückten Status pro Workspace. Ein Alarm kann durch Stummscha lten oder Unterdrückung unterbunden werden. Einheiten: Anzahl Gültige Statistiken: Durchschnitt, Minimum, Maximum

CloudWatch Name der Metrik	Ressource nname	CloudWatch Namespace	Description
ResourceCount	UnprocessedAlerts	AWS/Usage	Anzahl der Alarme im unbearbeiteten Zustand pro Workspace. Eine Warnung befindet sich im Status „Unverarbeitet“, sobald sie von empfangen wurde AlertManager, wartet aber auf die nächste Auswertung der Aggregationsgruppe. Einheiten: Anzahl Gültige Statistiken: Durchschnitt, Minimum, Maximum
ResourceCount	AllAlerts	AWS/Usage	Anzahl der Benachrichtigungen in einem beliebigen Status pro Workspace Einheiten: Anzahl Gültige Statistiken: Durchschnitt, Minimum, Maximum

CloudWatch Name der Metrik	Ressource nname	CloudWatch Namespace	Description
ResourceCount	AllRules	AWS/Usage	Anzahl der Regeln in einem beliebige n Bundesstaat pro Workspace Einheiten: Anzahl Gültige Statistiken: Durchschnitt, Minimum, Maximum
ResourceCount	NativeHis togramAct iveSeries	AWS/Usage	Die Anzahl der einzelnen aktiven nativen Histogram mserien pro Workspace Einheiten: Anzahl Gültige Statistiken: Durchschnitt, Minimum, Maximum
ResourceCount	NativeHis togramIng estionRate	AWS/Usage	Aufnahmerate für native Histogrammproben pro Workspace pro Sekunde Einheiten: Anzahl pro Sekunde Gültige Statistiken: Durchschnitt, Minimum, Maximum

CloudWatch Name der Metrik	Ressource nname	CloudWatch Namespace	Description
ActiveSeriesPerLabelSet	-	`AWS/Prometheus`	Die aktuelle Nutzung der aktiven Serie für jeden benutzerdefinierten Labelsatz Einheiten: Anzahl Gültige Statistiken: Minimum, Maximum, Summe, Durchschnitt
ActiveSeriesLimitPerLabelSet	-	`AWS/Prometheus`	Der aktuelle aktive Seriengrenzwert für jeden benutzerdefinierten Labelsatz Einheiten: Anzahl Gültige Statistiken: Minimum, Maximum, Summe, Durchschnitt
AlertManagerAlerts Received	-	`AWS/Prometheus`	Gesamtzahl der erfolgreichen Alerts, die vom Alert Manager empfangen wurden Einheiten: Anzahl Gültige Statistiken: Minimum, Maximum, Summe, Durchschnitt

CloudWatch Name der Metrik	Ressource nname	CloudWatch Namespace	Description
AlertManagerNotificationsFailed	-	`AWS/Prometheus`	Anzahl der fehlgeschlagenen Alarme Einheiten: Anzahl Gültige Statistiken: Minimum, Maximum, Summe, Durchschnitt
AlertManagerNotificationsThrottled	-	`AWS/Prometheus`	Anzahl der gedrosselten Alarme Einheiten: Anzahl Gültige Statistiken: Minimum, Maximum, Summe, Durchschnitt
AnomalyDetectors	WorkspaceId	`AWS/Prometheus`	Gesamtzahl der Anomaliedetektoren für einen bestimmten Arbeitsbereich Einheiten: Anzahl Gültige Statistiken: Durchschnitt, Minimum, Maximum
AnomalyDetectorEvaluations	WorkspaceId, AnomalyDetectorId	`AWS/Prometheus`	Gesamtzahl der Bewertungen durch Anomaliedetektoren Einheiten: Anzahl Gültige Statistiken: Minimum, Maximum, Summe, Durchschnitt

CloudWatch Name der Metrik	Ressource nname	CloudWatch Namespace	Description
AnomalyDetectorEvaluationFailures	WorkspaceId, AnomalyDetectorId	`AWS/Prometheus`	Anzahl der Ausfälle des Anomaliedetektors im Intervall Einheiten: Anzahl Gültige Statistiken: Minimum, Maximum, Summe, Durchschnitt
AnomalyDetectorLastEvaluationDuration	WorkspaceId, AnomalyDetectorId	`AWS/Prometheus`	Dauer der letzten Bewertung eines Anomaliedetektors Einheiten: Sekunden Gültige Statistiken: Minimum, Maximum, Summe, Durchschnitt
AnomalyDetectorMissedEvaluations	WorkspaceId, AnomalyDetectorId	`AWS/Prometheus`	Anzahl der verpassten Bewertungen durch einen Anomaliedetektor innerhalb des Intervalls Einheiten: Anzahl Gültige Statistiken: Minimum, Maximum, Summe, Durchschnitt

CloudWatch Name der Metrik	Ressource nname	CloudWatch Namespace	Description
Discarded Samples**	-	AWS/Prometheus	Anzahl der verworfenen Beispiele nach Grund Einheiten: Anzahl Gültige Statistiken: Minimum, Maximum, Summe, Durchschnitt
Discarded Series**	-	AWS/Prometheus	Anzahl der Serien, die eine verworfene Probe enthalten, aufgeschlüsselt nach dem Grund Einheiten: Anzahl Gültige Statistiken: Minimum, Maximum, Summe, Durchschnitt
Discarded SamplesPerLabelSet	-	AWS/Prometheus	Die Anzahl der verworfen en Proben für jeden benutzerdefinierten Labelsatz Einheiten: Anzahl Gültige Statistiken: Minimum, Maximum, Summe, Durchschnitt

CloudWatch Name der Metrik	Ressource nname	CloudWatch Namespace	Description
Discarded SeriesPer LabelSet	-	`AWS/Prometheus`	Die Anzahl der Serien, die eine verworfene Probe für jeden benutzerdefinierten Labelsatz enthalten Einheiten: Anzahl Gültige Statistiken: Minimum, Maximum, Summe, Durchschnitt
Ingestion RatePerLabelSet	-	`AWS/Prometheus`	Die Aufnahmerate für jeden benutzerdefinierten Etikettensatz Einheiten: Anzahl Gültige Statistiken: Minimum, Maximum, Summe, Durchschnitt
NativeHis togramIng estedBuck etsRate	-	`AWS/Prometheus`	Rate der pro Sekunde empfangenen befüllten Buckets in allen nativen Histogrammproben. Schließt abgelehnte Buckets aus. Einheiten: Anzahl pro Sekunde Gültige Statistiken: Durchschnitt, Minimum, Maximum

CloudWatch Name der Metrik	Ressource nname	CloudWatch Namespace	Description
NativeHistogramReducedResolutionCount	-	`AWS/Prometheus`	Anzahl der nativen Histogrammproben mit automatisch reduzierter Bucket-Auflösung. Die Auflösung wird reduziert, wenn eine Probe den Grenzwert für die maximale Anzahl an Buckets überschreitet. Einheiten: Anzahl Gültige Statistiken: Minimum, Maximum, Summe, Durchschnitt
OutOfOrderIngestionRate	-	`AWS/Prometheus`	Out-of-order Aufnahmerate der Probe Einheiten: Anzahl pro Sekunde Gültige Statistiken: Minimum, Maximum, Summe, Durchschnitt

CloudWatch Name der Metrik	Ressource nname	CloudWatch Namespace	Description
OutOfOrderSampleAge***	-	`AWS/Prometheus`	Die Differenz zwischen dem Zeitstempel einer Probe, die nicht in der richtigen Reihenfolge ist, und der Aufnahmezeit, die angibt, wie alt die Probe bei der Einnahme ist. Einheiten: Sekunden Gültige Statistiken: Minimum, Maximum, Summe, Durchschnitt
QuerySamplesProcessed	-	`AWS/Prometheus`	Anzahl der verarbeiteten Abfrageproben Einheiten: Anzahl Gültige Statistiken: Minimum, Maximum, Summe, Durchschnitt
RuleEvaluations	-	`AWS/Prometheus`	Gesamtzahl der Regelauswertungen Einheiten: Anzahl Gültige Statistiken: Minimum, Maximum, Summe, Durchschnitt

CloudWatch Name der Metrik	Ressource nname	CloudWatch Namespace	Description
RuleEvaluationFailures	-	`AWS/Prometheus`	Anzahl der Fehler bei der Regelauswertung im Intervall Einheiten: Anzahl Gültige Statistiken: Minimum, Maximum, Summe, Durchschnitt
RuleGroupIterationsMissed	-	`AWS/Prometheus`	Anzahl der Regelgruppen-Iterationen, die im Intervall verpasst wurden. Einheiten: Anzahl Gültige Statistiken: Minimum, Maximum, Summe, Durchschnitt
RuleGroupLastEvaluationDuration	-	`AWS/Prometheus`	Dauer der letzten Auswertung einer Regelgruppe. Einheiten: Sekunden Gültige Statistiken: Minimum, Maximum, Summe, Durchschnitt

* TPS-Metriken werden jede Minute generiert und sind ein Durchschnittswert pro Sekunde für diese Minute. Kurze Burst-Perioden werden in den TPS-Metriken nicht erfasst.

** Einige der Gründe, die dazu führen, dass Proben verworfen werden, sind folgende. Nicht alle unten aufgeführten Gründe sind in der DiscardedSeries Metrik enthalten.

Grund	Bedeutung
greater_than_max_sample_age	Proben, die älter als eine Stunde sind, werden verworfen.
new-value-for-timestamp	Doppelte Proben werden mit demselben Zeitstempel wie die vorherige Probe, aber mit unterschiedlichen Werten gesendet.
per_labelset_series_limit	Der Benutzer hat das Limit für die Gesamtzahl der aktiven Serien pro Labelsatz erreicht.
per_metric_series_limit	Der Benutzer hat das Limit für aktive Serien pro Metrik erreicht.
per_user_series_limit	Der Benutzer hat das Limit für die Gesamtzahl der aktiven Serien erreicht.
rate_limited	Die Aufnahmerate ist begrenzt.
sample-out-of-order	Die Proben werden nicht in der richtigen Reihenfolge versandt und können nicht bearbeitet werden.
Die Probe ist zu alt	Die Probe ist älter als das konfigurierte Zeitfenster, in dem die Reihenfolge abgelaufen ist, und kann nicht verarbeitet werden.
Ratenlimit bei Nichtbestellung	Out-of-order Die maximale Aufnahmerate wurde erreicht und die Probe kann nicht verarbeitet werden. Weitere Informationen finden Sie unter Servicekontingente von Amazon Managed Service für Prometheus.
label_value_too_long	Der Labelwert ist länger als die zulässige Zeichenbe schränkung.
max_label_names_per_series	Der Benutzer hat die Labelnamen pro Metrik eingegeben.
missing_metric_name	Der Metrikname wurde nicht angegeben.
metric_name_invalid	Ungültiger Metrikname angegeben.

Grund	Bedeutung
label_invalid	Ungültiges Etikett angegeben.
duplicate_label_names	Doppelte Labelnamen wurden angegeben.
native_histogram_sample_size_bytes_exceeded	Die native Histogrammprobe überschreitet die maximal zulässige Stichprobengröße in Byte.
native_histogram_invalid_schema	Das native Histogramm hat einen ungültigen Schemawert. Gültige Schemas liegen im Bereich von -4 bis 8.
native_histogram_invalid	Das native Histogramm schlägt bei der Überprüfung fehl (z. B. negative Bucket-Anzahl, nicht übereinstimmende Bucket-Anzahl oder falsch formatierte Bucket-Spans).
native_histogram_buckets_exceeded	Das native Histogramm überschreitet die maximale Anzahl an Buckets und die Auflösung kann nicht automatisch reduziert werden.
native_histogram_rate_limited	Eine native Histogrammprobe wurde zurückgewiesen, da die maximale Aufnahmerate für native Histogramme erreicht wurde.
per_user_native_histogram_series_limit	Der Benutzer hat das Limit für aktive Serien für native Histogramme pro Workspace erreicht.

*** Die `OutOfOrderSampleAge` Metrik beinhaltet eine `Percentile` Dimension. Anhand dieser Kennzahl können Sie das für Ihren Workspace passende Zeitfenster für Außerbetriebnahme ermitteln. Die gültigen Werte für die `Percentile` Dimension lauten wie folgt.

Perzentil	Description
p50	Das 50. Perzentilalter von Proben, die nicht in Ordnung sind.
p99	Das 99. Perzentil-Alter von Proben, die nicht in Ordnung sind.

Perzentil	Description
max	Das maximale Alter von Proben, die nicht in der richtigen Reihenfolge sind.

> **Note**
>
> Eine Metrik, die nicht existiert oder fehlt, entspricht dem Wert dieser Metrik, der 0 ist.

> **Note**
>
> `RuleGroupIterationsMissed`, `RuleEvaluationsRuleEvaluationFailures`, und `RuleGroupLastEvaluationDuration` haben die `RuleGroup` Dimension der folgenden Struktur:
> *`RuleGroupNamespace`*;*`RuleGroup`*

CloudWatch Alarm schlagen bei den von Prometheus verkauften Metriken

Sie können die Nutzung der Prometheus-Ressourcen mithilfe CloudWatch von Alarmen überwachen.

Um einen Alarm für die Anzahl von einzustellen ActiveSeries in Prometheus

1. Wählen Sie den Tab Graphed Metrics und scrollen Sie nach unten zum Label. ActiveSeries

   In der Ansicht Graphische Metriken werden nur die Metriken angezeigt, die gerade erfasst werden.
2. Wählen Sie das Benachrichtigungssymbol in der Spalte Aktionen aus.
3. Geben Sie unter Metrik und Bedingungen angeben die Schwellenwertbedingung in das Feld Bedingungswert ein und klicken Sie auf Weiter.
4. Wählen Sie unter Aktionen konfigurieren ein vorhandenes SNS-Thema aus oder erstellen Sie ein neues SNS-Thema, an das die Benachrichtigung gesendet werden soll.
5. In Namen und Beschreibung hinzufügen den Namen des Alarms und eine optionale Beschreibung hinzufügen.
6. Wählen Sie Alarm erstellen aus.

Überwachen Sie Amazon Managed Service für Prometheus-Ereignisse mit Protokollen CloudWatch

Amazon Managed Service for Prometheus protokolliert Fehler- und Warnereignisse von Alert Manager und Ruler in Protokollgruppen in Amazon CloudWatch Logs. Weitere Informationen zu Alert Manager und Rulers finden Sie im Thema Alert Manager in diesem Handbuch. Sie können die Workspace-Protokolldaten veröffentlichen, um Streams in CloudWatch Logs zu protokollieren. Sie können die Protokolle, die Sie überwachen möchten, in der Konsole für Amazon Managed Service für Prometheus oder mithilfe von AWS CLI konfigurieren. Sie können diese Protokolle in der CloudWatch Konsole anzeigen oder abfragen. Weitere Informationen zum Anzeigen von CloudWatch Log-Log-Streams in der Konsole finden Sie CloudWatchim CloudWatch Benutzerhandbuch unter Arbeiten mit Protokollgruppen und Log-Streams.

Das CloudWatch kostenlose Kontingent ermöglicht die Veröffentlichung von Protokollen bis zu 5 GB in CloudWatch Logs. Die Protokolle, die das im Rahmen des kostenlosen Kontingents zulässige Höchstmenge überschreiten, werden auf der Grundlage des CloudWatch Preisplans berechnet.

Themen

- CloudWatch Protokolle konfigurieren

CloudWatch Protokolle konfigurieren

Amazon Managed Service for Prometheus protokolliert Fehler- und Warnereignisse von Alert Manager und Ruler in Protokollgruppen in Amazon CloudWatch Logs.

Sie können die Konfiguration der CloudWatch Protokollprotokollierung in der Amazon Managed Service for Prometheus-Konsole oder in der festlegen, AWS CLI indem Sie die `create-logging-configuration` API-Anfrage aufrufen.

Voraussetzungen

Fügen Sie vor dem Aufrufen der ID oder Rolle`create-logging-configuration`, die Sie zur Konfiguration CloudWatch von Logs verwenden möchten, die folgende Richtlinie oder gleichwertige Berechtigungen bei.

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": "*"
        }
    ]
}
```

Um CloudWatch Logs zu konfigurieren

Sie können die Protokollierung in Amazon Managed Service for Prometheus entweder über die AWS Konsole oder die konfigurieren. AWS CLI

Console

Konfiguration der Protokollierung in der Konsole von Amazon Managed Service für Prometheus

1. Navigieren Sie in Ihrem Workspace-Detailbereich zur Registerkarte Logs.
2. Wählen Sie oben rechts im Bereich Protokolle die Option Protokolle verwalten aus.
3. Wählen Sie in der Drop-down-Liste „Protokollebene“ die Option „Alle“ aus.
4. Wählen Sie in der Dropdownliste Protokollgruppe die Protokollgruppe aus, in der Sie Ihre Protokolle veröffentlichen möchten.

   Sie können auch eine neue Protokollgruppe in CloudWatch der Konsole erstellen.
5. Wählen Sie Änderungen speichern aus.

AWS CLI

Sie können die Protokollierungskonfiguration mit dem festlegen AWS CLI.

Um die Protokollierung zu konfigurieren, verwenden Sie AWS CLI

- Führen Sie mit dem AWS CLI den folgenden Befehl aus.

```
aws amp create-logging-configuration --workspace-id my_workspace_ID
                                     --log-group-arn my-log-group-arn
```

Einschränkungen

- Nicht alle Ereignisse wurden protokolliert

  Amazon Managed Service für Prometheus protokolliert nur Ereignisse auf `warning`- oder `error`-Ebene.

- Richtlinie Größenbeschränkung

  CloudWatch Die Ressourcenrichtlinien für Protokolle sind auf 5120 Zeichen begrenzt. Wenn CloudWatch Logs feststellt, dass sich eine Richtlinie dieser Größenbeschränkung nähert, werden automatisch Protokollgruppen aktiviert, die mit `/aws/vendedlogs/` beginnen.

  Wenn Sie eine Warnungsregel mit aktivierter Protokollierung erstellen, muss Amazon Managed Service for Prometheus Ihre CloudWatch Logs-Ressourcenrichtlinie mit der von Ihnen angegebenen Protokollgruppe aktualisieren. Um zu verhindern, dass die Größenbeschränkung für die CloudWatch Logs-Ressourcenrichtlinie erreicht wird, stellen Sie Ihren CloudWatch Logs-Protokollgruppennamen ein Präfix voran. `/aws/vendedlogs/` Wenn Sie eine Protokollgruppe in der Konsole von Amazon Managed Service für Prometheus erstellen, wird den Namen der Protokollgruppen ein Präfix mit `/aws/vendedlogs/` vorangestellt. Weitere Informationen finden Sie im CloudWatch Logs-Benutzerhandbuch unter Aktivieren der Protokollierung von bestimmten AWS Diensten aus.

Verwaltung der Abfragekosten in Amazon Managed Service für Prometheus

Amazon Managed Service for Prometheus bietet die Möglichkeit, die Abfragekosten zu begrenzen, indem begrenzt wird, wie viele Query Samples Processed (QSP) von einer einzelnen Abfrage verwendet werden können. Sie können zwei Arten von Schwellenwerten für QSP konfigurieren: Warnung und Fehler, um die Abfragekosten effektiv verwalten und kontrollieren zu können.

Wenn Abfragen den Warnschwellenwert erreichen, wird in der API-Abfrageantwort eine Warnmeldung angezeigt. Bei Anfragen, die über Amazon Managed Grafana angezeigt werden, wird die Warnung in der Amazon Managed Grafana-Benutzeroberfläche angezeigt, sodass Benutzer teure Abfragen identifizieren können. Anfragen, die den Fehlerschwellenwert erreichen, werden nicht berechnet und mit einem Fehler zurückgewiesen.

Zusätzlich zur Abfragedrosselung bietet Amazon Managed Service for Prometheus die Möglichkeit, Abfrageleistungsdaten in Logs zu protokollieren. CloudWatch Mit dieser Funktion können Sie Anfragen detailliert analysieren und so Ihren Amazon Managed Service für Prometheus-Abfragen optimieren und die Kosten effektiver verwalten. Die Abfrageprotokollierung erfasst Informationen über Abfragen, die die angegebenen Schwellenwerte für die Verarbeitung von Query Samples (QSP) überschreiten. Diese Daten werden dann in CloudWatch Logs veröffentlicht, sodass Sie die Abfrageleistung untersuchen und analysieren können. Zu den protokollierten Abfragen gehören sowohl API-Abfragen als auch Regelabfragen. Standardmäßig ist die Abfrageprotokollierung deaktiviert, um die unnötige Nutzung von CloudWatch Protokollen zu minimieren. Sie können diese Funktion bei Bedarf für die Abfrageanalyse aktivieren.

Themen

- Konfiguration der Abfrageprotokollierung
- Schwellenwerte für die Abfragedrosselung konfigurieren
- Inhalt protokollieren
- Einschränkungen

Konfiguration der Abfrageprotokollierung

Sie können die Abfrageprotokollierung in der Amazon Managed Service for Prometheus-Konsole oder in der AWS CLI konfigurieren, indem Sie die `create-query-logging-configuration` API-Anfrage aufrufen. Dieser API-Text enthält eine Liste von Zielen, aber im Moment unterstützen wir

nur CloudWatch Logs als Ziel, und Ziele sollten genau ein Element mit CloudWatch Konfigurationen enthalten.

Voraussetzungen

Stellen Sie sicher, dass `logGroup` das bereits erstellt wurde. Die ID oder Rolle, die zur Konfiguration verwendet wurde, sollte über die folgenden Richtlinien oder gleichwertige Berechtigungen verfügen.

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": "*"
        }
    ]
}
```

CloudWatch Protokolle konfigurieren

Sie können CloudWatch Logs konfigurieren, indem Sie sich mit dem AWS-Managementkonsole oder dem bei Amazon Managed Service for Prometheus anmelden. AWS CLI

So konfigurieren Sie die Abfrageprotokollierung mit der Amazon Managed Service for Prometheus-Konsole

1. Navigieren Sie in Ihrem Workspace-Detailbereich zur Registerkarte Logs.
2. Wählen Sie unter Query Insights die Option Create aus.
3. Wählen Sie das Drop-down-Menü Protokollgruppe aus und wählen Sie die Protokollgruppe aus, um Ihre Protokolle zu veröffentlichen.

   Sie können auch eine neue Protokollgruppe in der CloudWatch Konsole erstellen.
4. Geben Sie den Schwellenwert (QSP) ein.
5. Wählen Sie Speichern.

Um die Abfrageprotokollierung mit dem AWS CLI Befehl zu konfigurieren

```
aws amp create-query-logging-configuration \
--workspace-id my_workspace_ID \
--destinations '[{"cloudWatchLogs":{"logGroupArn":"$my-log-group-arn"},"filters":
{"qspThreshold":$qspThreshold}}]'
```

Informationen zum Aktualisieren, Löschen und Beschreiben von Vorgängen finden Sie in der Amazon Managed Service for Prometheus API-Referenz.

Schwellenwerte für die Abfragedrosselung konfigurieren

Um QSP-Schwellenwerte zu konfigurieren, müssen Sie die Abfrageparameter in der API angeben. QueryMetrics

- max_samples_processed_warning_threshold — Legt den Warnschwellenwert für verarbeitete Abfrageproben fest
- max_samples_processed_error_threshold — Legt den Fehlerschwellenwert für verarbeitete Abfrageproben fest

Für Amazon Managed Grafana-Benutzer können Sie die Grafana-Datenquellenkonfiguration verwenden, um Beschränkungen auf alle Abfragen aus der Datenquelle anzuwenden:

1. Navigieren Sie zur Amazon Managed Service for Prometheus-Datenquellenkonfiguration in Amazon Managed Grafana.

2. Fügen Sie unter Benutzerdefinierte Abfrageparameter die Schwellenwert-Header hinzu.
3. Wählen Sie Speichern.

Inhalt protokollieren

Bei Abfragen, die auf Regeln basieren, werden in den CloudWatch Protokollen die folgenden Informationen zur Abfrage angezeigt:

```
{
    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"
}
```

Bei Abfragen, die aus API-Aufrufen stammen, werden in den CloudWatch Protokollen die folgenden Informationen zur Abfrage angezeigt:

```
{
    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"
}
```

Einschränkungen

Größenbeschränkungen für Richtlinien — Ressourcenrichtlinien für CloudWatch Protokolle sind auf 5120 Zeichen begrenzt. Wenn CloudWatch Logs feststellt, dass sich die Richtlinie der Größenbeschränkung nähert, werden automatisch Protokollgruppen aktiviert, die mit `/aws/vendedlogs/` beginnen. Wenn Sie die Abfrageprotokollierung aktivieren, muss Amazon Managed Service for Prometheus Ihre CloudWatch Logs-Ressourcenrichtlinie mit der von Ihnen angegebenen Protokollgruppe aktualisieren. Um zu verhindern, dass die Größenbeschränkung für die CloudWatch Logs-Ressourcenrichtlinie erreicht wird, stellen Sie Ihren CloudWatch Logs-Protokollgruppennamen das Präfix voran. `/aws/vendedlogs/`

Verstehen und optimieren Sie die Kosten in Amazon Managed Service für Prometheus

Die folgenden häufig gestellten Fragen und deren Antworten können hilfreich sein, um die mit Amazon Managed Service für Prometheus verbundenen Kosten zu verstehen und zu optimieren.

Was trägt zu meinen Kosten bei?

Für die meisten Kunden macht die Erfassung metrischer Daten den Großteil der Kosten aus. Bei Kunden, die häufig Abfragen nutzen, werden auch einige Kosten auf der Grundlage der verarbeiteten Abfragebeispiele anfallen, wobei die Speicherung von Metriken einen kleinen Teil der Gesamtkosten ausmacht. Weitere Informationen zu den jeweiligen Preisen finden Sie unter Preise auf der Produktseite für Amazon Managed Service für Prometheus.

Wie kann ich meine Kosten am besten senken? Wie senke ich die Kosten für die Erfassung?

Die Erfassungsraten (nicht die Speicherung der Messwerte) machen bei den meisten Kunden den Großteil der Kosten aus. Sie können die Erfassungsraten reduzieren, indem Sie die Erfassungshäufigkeit reduzieren (das Erfassungsintervall verlängern) oder indem Sie die Anzahl der aufgenommenen aktiven Serien reduzieren.

Sie können das Abholintervall (Scraping) von Ihrem Collection Agent erhöhen: Sowohl der Prometheus-Server (läuft im Agent-Modus) als auch der AWS Distro for OpenTelemetry (ADOT) Collector unterstützen die Konfiguration. `scrape_interval` Wenn Sie beispielsweise das Erfassungsintervall von 30 Sekunden auf 60 Sekunden erhöhen, wird Ihr Datenerfassungsverbrauch um die Hälfte reduziert.

Sie können die an Amazon Managed Service für Prometheus gesendeten Metriken auch mit `<relabel_config>` filtern. Weitere Informationen zur Neuetikettierung in der Prometheus-Agentenkonfiguration finden Sie unter https://prometheus.io/docs/prometheus/latest/configuration/configuration/#relabel_config in der Prometheus-Dokumentation.

Wie kann ich meine Abfragekosten am besten senken?

Die Abfragekosten basieren auf der Anzahl der verarbeiteten Beispiele. Sie können die Häufigkeit von Abfragen reduzieren, um Ihre Abfragekosten zu senken.

Weitere Informationen zu den Abfragen, die am meisten zu Ihren Abfragekosten beitragen, finden Sie unter. Verwaltung der Abfragekosten in Amazon Managed Service für Prometheus

Wenn ich die Aufbewahrungsdauer meiner Metriken verkürze, trägt das dazu bei, meine Gesamtrechnung zu reduzieren?

Sie können Ihre Aufbewahrungsfrist verkürzen, es ist jedoch unwahrscheinlich, dass dies Ihre Kosten wesentlich senkt.

Informationen zur Konfiguration der Aufbewahrungsdauer eines Workspace finden Sie unterKonfiguriere deinen Workspace.

Wie kann ich die Kosten für meine Warnungsabfragen niedrig halten?

Bei Benachrichtigungen werden Abfragen zu Ihren Daten erstellt, wodurch sich Ihre Abfragekosten erhöhen. Im Folgenden finden Sie einige Strategien, mit denen Sie Ihre Warnabfragen optimieren und Ihre Kosten senken können.

- Verwenden Sie Amazon Managed Service für Prometheus-Benachrichtigungen — Für Warnsysteme außerhalb von Amazon Managed Service for Prometheus sind möglicherweise zusätzliche Abfragen erforderlich, um die Stabilität oder Hochverfügbarkeit zu erhöhen, da der externe Service die Metriken aus mehreren Verfügbarkeitszonen oder Regionen abfragt. Dies beinhaltet Benachrichtigungen in Grafana für hohe Verfügbarkeit. Dies kann Ihre Kosten um das Dreifache oder mehr vervielfachen. Die Warnmeldungen in Amazon Managed Service for Prometheus sind optimiert und bieten Ihnen eine hohe Verfügbarkeit und Stabilität mit der geringsten Anzahl von Abfragen.

  Wir empfehlen, die systemeigenen Warnmeldungen in Amazon Managed Service for Prometheus anstelle von externen Warnsystemen zu verwenden.
- Optimieren Sie Ihr Warnintervall — Eine schnelle Möglichkeit, Ihre Warnabfragen zu optimieren, besteht darin, das Intervall für die automatische Aktualisierung zu verlängern. Wenn Sie eine

Warnung haben, die jede Minute abgefragt wird, aber nur alle fünf Minuten benötigt wird, können Sie durch eine Erhöhung des Intervalls für die automatische Aktualisierung das Fünffache Ihrer Abfragekosten für diese Warnung sparen.

- Verwenden Sie einen optimalen Lookback — Ein größeres Lookback-Fenster in Ihrer Abfrage erhöht die Kosten der Abfrage, da mehr Daten abgerufen werden. Stellen Sie sicher, dass das Lookback-Fenster in Ihrer PromQL-Abfrage eine angemessene Größe für die Daten hat, für die Sie eine Warnung ausgeben müssen. In der folgenden Regel umfasst der Ausdruck beispielsweise ein zehnminütiges Lookback-Fenster:

```
- alert: metric:alerting_rule
  expr: avg(rate(container_cpu_usage_seconds_total[10m])) > 0
  for: 2m
```

Wenn `avg(rate(container_cpu_usage_seconds_total[5m])) > 0` Sie `expr` den Wert ändern, können Sie Ihre Abfragekosten senken.

Schauen Sie sich im Allgemeinen Ihre Benachrichtigungsregeln an und stellen Sie sicher, dass Sie die für Ihren Service am besten geeigneten Messwerte verwenden. Es ist einfach, sich überschneidende Benachrichtigungen zu denselben Metriken oder mehrere Benachrichtigungen zu erstellen, die Ihnen dieselben Informationen liefern, insbesondere, wenn Sie im Laufe der Zeit Benachrichtigungen hinzufügen. Wenn Sie feststellen, dass häufig Gruppen von Benachrichtigungen gleichzeitig auftreten, ist es möglich, dass Sie Ihre Benachrichtigungen optimieren und nicht alle einbeziehen.

Diese Vorschläge können Ihnen helfen, die Kosten zu senken. Letztlich müssen Sie die Kosten mit der Erstellung der richtigen Warnmeldungen abwägen, um den Zustand Ihres Systems besser nachvollziehen zu können.

Weitere Informationen zu Warnmeldungen in Amazon Managed Service für Prometheus finden Sie unter. Verwaltung und Weiterleitung von Benachrichtigungen in Amazon Managed Service für Prometheus mit Alert Manager

Kann ich meine Rechnung jederzeit überprüfen?

Darin wird Ihre AWS Nutzung AWS Cost and Usage Report nachverfolgt und die geschätzten Gebühren für Ihr Konto innerhalb eines Abrechnungszeitraums angezeigt. Weitere Informationen

finden Sie unter Was sind AWS Kosten- und Nutzungsberichte? im Benutzerhandbuch AWS für Kosten- und Nutzungsberichte

Welche Metriken kann ich verwenden, um meine Kosten zu überwachen?

Metrische Proben, die Sie aufnehmen, sind der Hauptkostentreiber für Amazon Managed Service for Prometheus. Die Anzahl der aufgenommenen Proben bestimmt direkt Ihre monatlichen Gebühren. Daher ist es wichtig, Ihre Aufnahmemuster zu überwachen und zu verstehen.

AWS Cost Explorerist die Informationsquelle für die Überwachung Ihrer Amazon Managed Service for Prometheus-Kosten. Sie können den Cost Explorer für Amazon Managed Service for Prometheus auf historische und tägliche Kostentrends in verschiedenen Dimensionen, einschließlich aufgenommener Proben, überwachen.AWS Die Erkennung von Kostenanomalien bietet Ihnen auch die Möglichkeit, unerwartete Veränderungen Ihres Ausgabenverhaltens zu überwachen.

Die Verwendung von `IngestionRate` Kennzahlen bietet eine hilfreiche Methode zur Überwachung von Konsumtrends, die in direktem Zusammenhang mit den Kosten stehen. Die Verwendung `IngestionRate` als zusätzliche Kennzahl bietet unter anderem folgende Vorteile:

- Workspace-level Nachverfolgung — Überwachen Sie die Aufnahme pro Workspace und nicht nur auf Kontoebene.
- Detaillierte Transparenz — Verfolge die Aufnahmemuster auf stündlicher oder sogar minutenweiser Basis, um Einblicke in Echtzeit zu erhalten.
- Proaktive Überwachung — Richten Sie CloudWatch Alarme ein, um Nutzungsspitzen zu erkennen, bevor sie in der Abrechnung erscheinen.

> **Note**
>
> `IngestionRate`kann verwendet werden, um Kosten und Trends zu schätzen oder Kosten pro Workspace zuzuordnen, ist aber nicht zu 100% genau. Da es sich `IngestionRate` um eine Durchschnittsrate handelt, die in Intervallen von einer Minute abgetastet wird, ergibt die Multiplikation dieser Rate mit der Zeit eher eine Näherung als eine exakte Anzahl der aufgenommenen Proben. Darüber hinaus wirkt sich die Datenaufbewahrungsrichtlinie CloudWatch von Amazon auf die Granularität aus, die für historische Abfragen verfügbar ist, wobei Daten, die älter als 63 Tage sind, auf Intervalle von 1 Stunde begrenzt sind.

Weitere Informationen zur Überwachung von Amazon Managed Service for Prometheus-Metriken finden Sie unter CloudWatch. Verwenden Sie CloudWatch Metriken, um die Ressourcen von Amazon Managed Service for Prometheus zu überwachen

Überwachung systemeigener Histogrammkosten — Systemeigene Histogramme werden auf der Grundlage ausgefüllter Buckets berechnet: Jeder ausgefüllte Bucket (ein Bucket mit Beobachtungen ungleich Null) zählt als 0,25 einer metrischen Stichprobe, und leere Buckets werden nicht gemessen. Ein systemeigenes Histogramm mit 10 ausgefüllten Buckets wird beispielsweise als 2,5 metrische Stichproben gemessen. Um Ihre nativen Histogramm-Kostentreiber zu überwachen, können Sie die `NativeHistogramIngestedBucketsRate` Metrik verwenden, um zu überwachen, wie häufig aufgefüllte Bereiche aufgenommen wurden. Sie können sie auch verwenden`NativeHistogramIngestionRate`, um die Gesamtrate der systemeigenen Histogrammstichproben nachzuverfolgen und die Anzahl der aktiven nativen Histogrammreihen in Ihrem Workspace `NativeHistogramActiveSeries` zu überwachen.

Wie sehe ich meine Kosten in AWS Cost Explorer?

Als Informationsquelle für die Kosten von Amazon Managed Service für Prometheus AWS Cost Explorer bietet es Ihre tatsächlich in Rechnung gestellte Nutzung und die Gebühren für Amazon Managed Service für aufgenommene Prometheus-Proben, einschließlich historischer Abrechnungsdaten nach Monat und Region. Verwenden Sie den Cost Explorer für Ihre endgültigen Rechnungsbeträge und die täglichen Kostentrends.

So zeigen Sie Ihre Amazon Managed Service for Prometheus Kosten an:

Zugriff AWS Cost Explorer

1. Melden Sie sich bei der AWS Management Console an.
2. Navigieren Sie zum Abrechnungs Billing and Cost Management Kostenmanagement-Dashboard.
3. Wählen Sie im linken Navigationsmenü Cost Explorer aus.
4. Wählen Sie Cost Explorer starten (falls Sie ihn zum ersten Mal verwenden).

Konfigurieren Sie den Bericht

1. Stellen Sie Ihren Zeitraum auf den gewünschten Abrechnungszeitraum ein (z. B. März 2025 bis Februar 2026).
2. Wählen Sie unter Filter Folgendes aus:

- Service: Wählen Sie „Amazon Managed Service für Prometheus“.
- Verwendungsart: Filtern Sie nach "MetricSampleCount", um die Gebühren für die Probenaufnahme zu isolieren.

Daten gruppieren und anzeigen

1. Wählen Sie unter Gruppieren nach die Option Region aus, um Kosten- und Nutzungsdaten pro Region anzuzeigen.
2. Wählen Sie Ihre bevorzugte Visualisierung (Balkendiagramm, Liniendiagramm oder Tabelle).
3. Wählen Sie Anwenden, um den Bericht zu erstellen.

Daten exportieren (optional)

1. Wählen Sie in der oberen rechten Ecke CSV herunterladen, um die Daten zu exportieren.
2. Die CSV-Datei enthält: Abrechnungszeitraum, Region, Nutzungsart, abgerechneter Betrag und Nutzungsmenge (Anzahl der in Rechnung gestellten Proben).

> **Note**
>
> Cost Explorer Explorer-Daten weisen in der Regel eine Verzögerung von 24 Stunden auf. Für den aktuellsten Abrechnungszeitraum sind die Daten möglicherweise erst am nächsten Tag verfügbar.

Wie berechne ich die Anzahl der in einem Monat aufgenommenen Proben?

Sie können die ungefähre Anzahl der aufgenommenen Proben anhand der `IngestionRate` Kennzahlen CloudWatch von Amazon mit dem AWS Command Line Interface berechnen. Dies ist nützlich, um monatliche Rechnungen zu überprüfen und die Nutzungsmuster in den verschiedenen Arbeitsbereichen zu verstehen.

So rufen Sie Aufnahmedaten ab:

```
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"
            }
        }
    ]'
```

Der Befehl gibt stündliche `IngestionRate` Durchschnittswerte zurück, gemessen in Stichproben pro Sekunde. Um die ungefähre Anzahl der in einem Monat aufgenommenen Proben zu berechnen, multiplizieren Sie jeden stündlichen Datenpunkt mit 3600 (Sekunden pro Stunde), um die in dieser Stunde aufgenommenen Proben zu erhalten, und summieren Sie dann alle stündlichen Gesamtwerte für den Monat:

```
Monthly samples ≈ Σ (hourly IngestionRate average × 3600)
```

Wenn beispielsweise eine einzelne Stunde einen Durchschnitt `IngestionRate` von 500 Proben pro Sekunde ergibt, hat diese Stunde ungefähr 500 × 3600 = 1.800.000 Proben beigetragen. Wiederholen Sie diesen Vorgang für jede Stunde des Monats und summieren Sie die Ergebnisse, um Ihre ungefähre monatliche Aufnahmemenge zu erhalten.

Hauptparameter:

- `Period`: 3600 (1 Stunde in Sekunden)
- `StartTime`: Beginn Ihres Monats (zum Beispiel`2026-02-01T00:00:00Z`)
- `EndTime`: Ende Ihres Monats (zum Beispiel`2026-03-01T00:00:00Z`)
- `Stat`: Durchschnittlich

So findest du deine Workspace-IDs heraus:

```
aws amp list-workspaces --region your-region
```

Verwenden Sie die Workspace-ID, um die Metriken so zu filtern, dass nur Daten für den angegebenen Workspace angezeigt werden, anstatt sie für alle Prometheus-Ressourcen in der Region zu aggregieren.

Welche Datengranularität ist für die Analyse historischer Kosten verfügbar?

CloudWatchDie Datenaufbewahrungsrichtlinie von Amazon wirkt sich auf die Granularität aus, die für historische Abfragen verfügbar ist:

- Daten, die weniger als 15 Tage alt sind: Abfragen in Intervallen von 1 Minute (`Period`: 60)
- Daten, die 15—63 Tage alt sind: Abfrage in 5-Minuten-Intervallen (`Period`: 300)
- Daten, die älter als 63 Tage sind: Beschränkt auf Intervalle von 1 Stunde (`Period`: 3600)

Bei historischen Analysen, die länger als 63 Tage dauern, CloudWatch erfolgt ein automatisches Downsampling der Daten auf einen Zeitraum von mindestens einer Stunde. Wenn Sie die Abrechnung für Monate überprüfen, die älter als 63 Tage sind, müssen Sie stündlich aggregierte Daten verwenden. Bei der monatlichen Beispielberechnung werden diese stündlich gemittelten Datenpunkte verwendet, wobei jeder Wert im gesamten Monat mit 3600 multipliziert wird.

Diese geringere Granularität trägt weiter dazu bei, dass für ältere Daten Schätzungen statt exakte Zahlen bereitgestellt werden. Ihre verbindlichen Rechnungsbeträge finden Sie immer im Cost Explorer. `IngestionRate`

Weitere Informationen zur CloudWatch Aufbewahrung von Kennzahlen finden Sie unter Aufbewahrung von Kennzahlen im CloudWatch Amazon-Benutzerhandbuch.

Was sind bewährte Methoden für die Überwachung der Kosten von Amazon Managed Service for Prometheus?

Um Ihre Ausgaben für Amazon Managed Service for Prometheus effektiv zu verwalten und zu optimieren, sollten Sie die Implementierung der folgenden Überwachungspraktiken in Betracht ziehen:

- Überwachen Sie den Cost Explorer regelmäßig, um die tatsächlichen Ausgabentrends zu verfolgen und Kostenanomalien in verschiedenen Dimensionen, einschließlich aufgenommener Stichproben, zu identifizieren.
- Aktivieren Sie die Erkennung von AWS Kostenanomalien, um Benachrichtigungen über unerwartete Kostensteigerungen Ihrer Ausgaben für Amazon Managed Service for Prometheus zu erhalten.
- Schalten Sie CloudWatch Alarme ein, um die Überwachung auf `IngestionRate` Arbeitsplatzebene und die Früherkennung von Aufnahmespitzen zu ermöglichen.
- Exportieren Sie regelmäßig Cost Explorer Explorer-Daten für langfristige Kostenanalysen und Berichte.

Warum ist meine Rechnung zu Beginn des Monats höher als am Monatsende?

Amazon Managed Service für Prometheus hat ein gestaffeltes Preismodell für die Erfassung, was dazu führt, dass die Kosten bei der ersten Nutzung höher sind. Wenn Ihre Nutzung höhere Erfassungsstufen erreicht und die Kosten sinken, sinken auch Ihre Kosten. Weitere Informationen zu den Preisen, einschließlich der Erfassungsstufen, finden Sie auf der Produktseite von Amazon Managed Service für Prometheus unter Preise.

> **Note**
>
> - Stufen gelten für die Nutzung innerhalb einer Region, nicht für die Nutzung zwischen Regionen. Die Nutzung innerhalb einer Region muss die nächste Stufe erreichen, um den niedrigeren Tarif nutzen zu können.

- In einer Organisation innerhalb der AWS Organizations Stufe wird die Nutzung pro Zahlerkonto und nicht pro Konto berechnet (das Zahlerkonto ist immer das Verwaltungskonto der Organisation). Wenn die Gesamtzahl der erfassten Kennzahlen (innerhalb einer Region) für alle Konten in einer Organisation die nächste Stufe erreicht, wird allen Konten der niedrigere Tarif berechnet.

Ich habe alle meine Amazon Managed Service for Prometheus-Arbeitsbereiche gelöscht, aber es scheint, dass mir immer noch Gebühren berechnet werden. Was könnte passieren?

In diesem Fall besteht eine Möglichkeit darin, dass Sie immer noch AWS verwaltete Scraper haben, die so eingerichtet sind, dass sie Metriken an Ihre gelöschten Workspaces senden. Folgen Sie den Anweisungen zu. Suchen und Löschen von Scrapern

Integration mit anderen AWS Diensten

Amazon Managed Service für Prometheus lässt sich in andere AWS Dienste integrieren. In diesem Abschnitt wird die Integration der Kostenüberwachung mit Amazon Elastic Kubernetes Service (Amazon EKS) (mit Kubecost) und die Erfassung von Metriken aus CloudWatch Amazon Data Firehose beschrieben. Es beschreibt auch die Einrichtung und Verwaltung von Amazon Managed Service für Prometheus mit AWS Observability Accelerator Terraform-Modulen oder mithilfe von Controllern für Kubernetes. AWS

Themen

- Integration in Amazon-EKS-Kostenüberwachung
- Amazon Managed Service für Prometheus mit AWS Observability Accelerator einrichten
- Amazon Managed Service für Prometheus mit AWS Controllern für Kubernetes verwalten
- Integration von CloudWatch Metriken mit Amazon Managed Service für Prometheus

Integration in Amazon-EKS-Kostenüberwachung

Amazon Managed Service für Prometheus ist in die Kostenüberwachung von Amazon Elastic Kubernetes Service (Amazon EKS) (mit Kubecost) integriert, um Berechnungen zur Kostenzuweisung durchzuführen und Einblicke in die Optimierung Ihrer Kubernetes-Cluster zu erhalten. Wenn Sie Amazon Managed Service für Prometheus mit Kubecost verwenden, können Sie Ihre Kostenüberwachung zuverlässig skalieren, um größere Cluster zu unterstützen.

Durch die Integration mit Kubecost erhalten Sie einen detaillierten Überblick über Ihre Amazon-EKS-Clusterkosten. Sie können die Kosten nach den meisten Kubernetes-Kontexten zusammenfassen, von der Container-Ebene über die Cluster-Ebene bis hin zur Multi-Cluster-Ebene. Sie können container- oder clusterübergreifende Berichte erstellen, um die Kosten für Rückbuchungen nachzuverfolgen.

Im Folgenden finden Sie Anweisungen zur Integration mit Kubecost in einem Einzel- oder Multi-Cluster-Szenario:

- Single-Cluster-Integration – Informationen zur Integration der Amazon-EKS-Kostenüberwachung in einen einzelnen Cluster finden Sie im AWS -Blogbeitrag Kubecost in Amazon Managed Service für Prometheus integrieren.

- Multi-Cluster-Integration – Informationen zur Integration der Amazon-EKS-Kostenüberwachung in mehrere Cluster finden Sie im AWS -Blogbeitrag Multi-Cluster-Kostenüberwachung für Amazon EKS mithilfe von Kubecost und Amazon Managed Service für Prometheus.

> **Note**
>
> Weitere Informationen zur Verwendung von Kubecost finden Sie unter Kostenüberwachung im Amazon-EKS-Benutzerhandbuch.

Amazon Managed Service für Prometheus mit AWS Observability Accelerator einrichten

AWS bietet Observability-Tools, einschließlich Überwachung, Protokollierung, Warnmeldungen und Dashboards, für Ihre Amazon Elastic Kubernetes Service (Amazon EKS) -Projekte. Dazu gehören Amazon Managed Service für Prometheus, Amazon Managed Grafana, AWS Distro for OpenTelemetry und andere Tools. Um Ihnen zu helfen, diese Tools zusammen zu verwenden, AWS bietet Terraform-Module, die Observability mit diesen Diensten konfigurieren, den sogenannten Observability Accelerator.AWS

AWS Observability Accelerator bietet zwei Collector-Profile für Amazon Managed Service for Prometheus:

- Verwaltete Metriken (agentenlos) — Verwendet den Amazon Managed Service for Prometheus Collector, einen vollständig verwalteten, agentenlosen Scraper, der außerhalb Ihres Clusters ausgeführt wird. Es müssen keine Collector-Pods verwaltet werden. Nur Metriken.
- Selbstverwaltet — Stellt einen OpenTelemetry Collector über Helm in Ihrem Cluster bereit. Unterstützt Metriken, Traces (AWS X-Ray) und Logs (Amazon CloudWatch).

In diesem Abschnitt werden beide Optionen beschrieben, beginnend mit dem empfohlenen agentenlosen Ansatz.

Die Terraform-Vorlagen und detaillierte Anweisungen finden Sie auf der Seite AWS Observability Accelerator for Terraform. GitHub

Voraussetzungen

Um AWS Observability Accelerator verwenden zu können, müssen Sie über einen vorhandenen Amazon EKS-Cluster verfügen und die folgenden Voraussetzungen erfüllen:

- AWS CLI— wird verwendet, um AWS Funktionen von der Befehlszeile aus aufzurufen.
- kubectl— wird verwendet, um Ihren EKS-Cluster von der Befehlszeile aus zu steuern.
- Terraform (>= 1.5.0) — wird verwendet, um die Erstellung der Ressourcen für diese Lösung zu automatisieren. Sie müssen für den AWS Anbieter eine IAM-Rolle eingerichtet haben, die Zugriff auf die Erstellung und Verwaltung von Amazon Managed Service for Prometheus, Amazon Managed Grafana und IAM innerhalb Ihres Kontos hat. AWS Weitere Informationen zur Konfiguration des AWS Anbieters für Terraform finden Sie unter AWS Anbieter in der Terraform-Dokumentation.

Verwenden Sie das Beispiel für verwaltete Metriken (ohne Agenten)

In diesem Beispiel wird der Amazon Managed Service for Prometheus Collector verwendet, um Prometheus-Metriken aus Ihrem Amazon EKS-Cluster zu extrahieren, ohne Collector-Pods bereitzustellen. Der Collector benötigt mindestens zwei Subnetze in zwei unterschiedlichen Availability Zones. Weitere Einzelheiten finden Sie im eks-amp-managedBeispiel unter GitHub.

Um das Terraform-Modul zur agentenlosen Infrastrukturüberwachung zu verwenden

1. Klonen Sie das Repo aus dem Ordner, in dem Sie Ihr Projekt erstellen möchten, mit dem folgenden Befehl.

   ```
   git clone https://github.com/aws-observability/terraform-aws-observability-accelerator.git
   ```

2. Initialisieren Sie Terraform mit den folgenden Befehlen.

   ```
   cd examples/eks-amp-managed

   terraform init
   ```

3. Erstellen Sie eine neue `terraform.tfvars` Datei wie im folgenden Beispiel. Verwenden Sie die AWS Region, Cluster-ID und VPC-Netzwerkdetails für Ihren Amazon EKS-Cluster. Der Collector benötigt mindestens zwei Subnetze in zwei unterschiedlichen Availability Zones.

```
# (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. Erstellen Sie einen Workspace in Amazon Managed Grafana, wenn Sie noch keinen haben, den Sie verwenden möchten. Informationen zum Erstellen eines neuen Workspace finden Sie unter Erstellen Sie Ihren ersten Workspace im Benutzerhandbuch von Amazon Managed Grafana.
5. Erstellen Sie zwei Variablen für Terraform, um Ihren Grafana-Workspace zu verwenden, indem Sie die folgenden Befehle in der Befehlszeile ausführen. Sie müssen das durch die *grafana-workspace-id* ID aus Ihrem Grafana-Workspace ersetzen.

```
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. [Optional] Um einen vorhandenen Amazon Managed Service for Prometheus Workspace zu verwenden, fügen Sie die ID wie im folgenden Beispiel zur `terraform.tfvars` Datei hinzu und ersetzen Sie die durch Ihre Prometheus-Workspace-ID. *prometheus-workspace-id* Wenn Sie keinen vorhandenen Workspace angeben, wird ein neuer Prometheus-Workspace für Sie erstellt.

```
# (optional) Leave it empty for a new workspace to be created
managed_prometheus_workspace_id = "prometheus-workspace-id"
```

7. Stellen Sie die Lösung mit dem folgenden Befehl bereit.

```
terraform apply -var-file=terraform.tfvars
```

Dadurch werden Ressourcen in Ihrem AWS Konto erstellt, darunter die folgenden:

- Ein neuer Workspace in Amazon Managed Service für Prometheus (sofern Sie sich nicht für die Nutzung eines vorhandenen Workspace entschieden haben).
- Ein Amazon Managed Service for Prometheus Collector (agentenloser Scraper), der so konfiguriert ist, dass er Prometheus-Metriken aus Ihrem Amazon EKS-Cluster ausliest.
- Prometheus-Aufzeichnungs- und Warnregeln in Ihrem Amazon Managed Service for Prometheus-Workspace.
- kube-state-metrics und Node-Exporter, der in Ihrem Amazon EKS-Cluster für Infrastrukturmetriken bereitgestellt wird.
- Neue Amazon-Managed-Grafana-Datenquelle und Dashboards in Ihrem aktuellen Workspace. Die Dashboards werden unter EKS Monitoring aufgeführt.

Alternative: Selbstverwalteter Collector OpenTelemetry

Wenn Sie Ablaufverfolgungen, Protokolle oder die vollständige Kontrolle über die Erfassungspipeline benötigen, verwenden Sie das selbstverwaltete Profil. Dadurch wird ein OpenTelemetry Collector über Helm in Ihrem Amazon EKS-Cluster bereitgestellt, der so konfiguriert ist, dass er Prometheus-Metriken scannt und remote in Amazon Managed Service for Prometheus schreibt. Es unterstützt auch Traces (AWS X-Ray) und Logs (Amazon CloudWatch). Weitere Einzelheiten finden Sie im eks-amp-otelBeispiel unter GitHub.

Um das selbstverwaltete Terraform-Modul zu verwenden

1. Klonen Sie das Repo und initialisieren Sie Terraform.

```
git clone https://github.com/aws-observability/terraform-aws-observability-
accelerator.git
cd examples/eks-amp-otel
terraform init
```

2. Erstellen Sie eine neue `terraform.tfvars` Datei wie im folgenden Beispiel.

```
# (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. Richten Sie Ihren Amazon Managed Grafana-Workspace und Ihren API-Schlüssel mit denselben Schritten ein wie im Beispiel für verwaltete Metriken (Schritte 4—6 oben).
4. Stellen Sie die Lösung mit dem folgenden Befehl bereit.

```
terraform apply -var-file=terraform.tfvars
```

Dadurch werden die folgenden Ressourcen in Ihrem AWS Konto erstellt (im Gegensatz zum agentenlosen Ansatz wird der Collector in Ihrem Cluster ausgeführt):

- Ein Amazon Managed Service für Prometheus Workspace (falls nicht bereitgestellt).
- Ein Amazon Managed Grafana-Arbeitsbereich mit Datenquelle und Dashboards.
- Ein über Helm in Ihrem Amazon EKS-Cluster bereitgestellter OpenTelemetry Collector, der so konfiguriert ist, dass er Prometheus-Metriken scannt und remote in Amazon Managed Service for Prometheus schreibt.
- Eine IAM-Rolle für Dienstkonten (IRSA) für den Collector. OpenTelemetry
- Verfolgt die Pipeline zu AWS X-Ray (standardmäßig aktiviert).
- Protokolliert die Pipeline zu Amazon CloudWatch (standardmäßig aktiviert).

Dashboards anzeigen

Um Ihre neuen Dashboards anzuzeigen, öffnen Sie das entsprechende Dashboard in Ihrem Workspace in Amazon Managed Grafana. Die Infrastruktur-Dashboards werden automatisch von Terraform bereitgestellt. Weitere Informationen zur Verwendung von Amazon Managed Grafana finden Sie unter Arbeiten in Ihrem Grafana-Workspace im Benutzerhandbuch von Amazon Managed Grafana.

Amazon Managed Service für Prometheus mit AWS Controllern für Kubernetes verwalten

Amazon Managed Service für Prometheus ist in AWS Controllers für Kubernetes (ACK) integriert und unterstützt die Verwaltung Ihrer Workspace-, Alert Manager- und Ruler-Ressourcen in Amazon EKS. Sie können benutzerdefinierte Ressourcendefinitionen (CRDs) von AWS Controllers for Kubernetes und native Kubernetes-Objekte verwenden, ohne Ressourcen außerhalb Ihres Clusters definieren zu müssen.

In diesem Abschnitt wird beschrieben, wie Sie AWS Controller für Kubernetes und Amazon Managed Service for Prometheus in einem vorhandenen Amazon EKS-Cluster einrichten.

Sie können auch die Blogbeiträge zur Einführung von AWS Controllern für Kubernetes und zur Einführung des ACK-Controllers für Amazon Managed Service for Prometheus lesen.

Voraussetzungen

Bevor Sie mit der Integration von AWS Controllern für Kubernetes und Amazon Managed Service for Prometheus in Ihren Amazon EKS-Cluster beginnen, müssen Sie die folgenden Voraussetzungen erfüllen.

- Sie benötigen ein vorhandenes Konto AWS-Konto und die erforderlichen Berechtigungen, um Amazon Managed Service for Prometheus- und IAM-Rollen programmgesteuert zu erstellen.
- Sie müssen über einen vorhandenen Amazon-EKS-Cluster mit aktiviertem OpenID Connect (OIDC) verfügen.

  Wenn Sie OIDC nicht aktiviert haben, können Sie es mit dem folgenden Befehl aktivieren. Denken Sie daran, das *YOUR_CLUSTER_NAME* und durch die richtigen Werte für *AWS_REGION* Ihr Konto zu ersetzen.

  ```
  eksctl utils associate-iam-oidc-provider \
      --cluster ${YOUR_CLUSTER_NAME} --region ${AWS_REGION} \
      --approve
  ```

  Weitere Informationen zur Verwendung von OIDC mit Amazon EKS finden Sie unter OIDC-Identitätsanbieter-Authentifizierung und Erstellen eines IAM-OIDC-Anbieters im Amazon-EKS-Benutzerhandbuch.
- Sie müssen den Amazon-EBS-CSI-Treiber in Ihrem Amazon-EKS-Cluster installiert haben.
- Sie müssen AWS CLI installiert haben. Das AWS CLI wird verwendet, um AWS Funktionen von der Befehlszeile aus aufzurufen.
- Helm, der Paketmanager für Kubernetes, muss installiert sein.
- Metriken der Steuerebene mit Prometheus müssen in Ihrem Amazon-EKS-Cluster eingerichtet werden.
- Sie müssen ein Amazon Simple Notification Service (Amazon SNS)-Thema haben, zu dem Sie Benachrichtigungen von Ihrem neuen Workspace aus senden möchten. Vergewissern Sie sich,

dass Sie Amazon Managed Service für Prometheus die Erlaubnis erteilt haben, Nachrichten zu diesem Thema zu senden.

Wenn Ihr Amazon-EKS-Cluster entsprechend konfiguriert ist, sollten Sie in der Lage sein, Metriken zu sehen, die für Prometheus formatiert sind, indem Sie `kubectl get --raw /metrics` aufrufen. Jetzt sind Sie bereit, einen AWS Controllers for Kubernetes Service Controller zu installieren und ihn zur Bereitstellung von Amazon Managed Service for Prometheus-Ressourcen zu verwenden.

Bereitstellung eines Workspace mit Controllern für Kubernetes AWS

Um einen neuen Amazon Managed Service for Prometheus Workspace bereitzustellen, installieren Sie einen AWS Controller für Controller für Kubernetes und verwenden diesen dann, um den Workspace zu erstellen.

Um einen neuen Amazon Managed Service für Prometheus Workspace mit AWS Controllern für Kubernetes bereitzustellen

1. Verwenden Sie die folgenden Befehle, um Helm zur Installation des Service Controllers von Amazon Managed Service für Prometheus zu verwenden. Weitere Informationen finden Sie unter Installieren eines ACK-Controllers in der Dokumentation zu AWS Controllern für Kubernetes unter. GitHub Verwenden Sie den richtigen *`region`* für Ihr System, z. B. `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
   ```

   Nach einigen Augenblicken sollten Sie eine Reaktion ähnlich der folgenden erhalten, die auf einen Erfolg hinweist.

   ```
   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"
```

Sie können optional mit dem folgenden Befehl überprüfen, ob der AWS Controller für Kubernetes Controller erfolgreich installiert wurde.

```
helm list --namespace $ACK_SYSTEM_NAMESPACE -o yaml
```

Dadurch werden Informationen über den Controller zurückgegeben`ack-prometheusservice-controller`, einschließlich des `status: deployed`.

2. Erstellen Sie eine Datei mit dem Namen `workspace.yaml` und folgendem Text. Dies wird als Konfiguration für den Workspace verwendet, den Sie erstellen.

```
apiVersion: prometheusservice.services.k8s.aws/v1alpha1
kind: Workspace
metadata:
  name: my-amp-workspace
spec:
  alias: my-amp-workspace
  tags:
    ClusterName: EKS-demo
```

3. Führen Sie den folgenden Befehl aus, um Ihren Workspace zu erstellen (dieser Befehl hängt von den Systemvariablen ab, die Sie in Schritt 1 eingerichtet haben).

```
kubectl apply -f workspace.yaml -n $ACK_SYSTEM_NAMESPACE
```

Innerhalb weniger Augenblicke sollten Sie in der Lage sein, einen neuen Workspace zu sehen, der `my-amp-workspace` in Ihrem Konto aufgerufen wird.

Führen Sie den folgenden Befehl aus, um die Details und den Status Ihres Workspace einschließlich der Workspace-ID anzuzeigen. Alternativ können Sie den neuen Workspace in der Konsole für Amazon Managed Service für Prometheus anzeigen.

```
kubectl describe workspace my-amp-workspace -n $ACK_SYSTEM_NAMESPACE
```

> **Note**
>
> Sie können auch einen vorhandenen Workspace verwenden, anstatt einen neuen zu erstellen.

4. Erstellen Sie zwei neue Yaml-Dateien als Konfiguration für die Rulegroups AlertManager , die Sie als Nächstes mit der folgenden Konfiguration erstellen werden.

   Speichern Sie diese Konfiguration unter `rulegroup.yaml`. Ersetzen Sie es *WORKSPACE-ID* durch die Workspace-ID aus dem vorherigen Schritt.

```
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 }}"
```

Speichern Sie die folgende Konfiguration als `alertmanager.yaml`. Ersetze es *`WORKSPACE-ID`* durch die Workspace-ID aus dem vorherigen Schritt. *`TOPIC-ARN`*Ersetzen Sie es durch den ARN für das Amazon SNS SNS-Thema, an das Sie Benachrichtigungen senden möchten, und *`REGION`* durch das, das AWS-Region Sie verwenden. Denken Sie daran, dass Amazon Managed Service für Prometheus über Berechtigungen für das Amazon-SNS-Thema verfügen muss.

```
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**
>
> Weitere Informationen zu den Formaten dieser Konfigurationsdateien finden Sie unter RuleGroupsNamespaceData und AlertManagerDefinitionData.

5. Führen Sie die folgenden Befehle aus, um Ihre Regelgruppen- und Alert-Manager-Konfiguration zu erstellen (dieser Befehl hängt von den Systemvariablen ab, die Sie in Schritt 1 eingerichtet haben).

```
kubectl apply -f rulegroup.yaml -n $ACK_SYSTEM_NAMESPACE
kubectl apply -f alertmanager.yaml -n $ACK_SYSTEM_NAMESPACE
```

Die Änderungen werden innerhalb von einigen Augenblicken verfügbar sein.

> **Note**
>
> Um eine Ressource zu aktualisieren, anstatt sie zu erstellen, aktualisieren Sie einfach die Yaml-Datei und führen den Befehl `kubectl apply` erneut aus.
> Um eine Ressource zu löschen, führen Sie den folgenden Befehl aus.
> *`ResourceType`*Ersetzen Sie durch den Ressourcentyp, den Sie löschen möchten `WorkspaceAlertManagerDefinition`, oder`RuleGroupNamespace`.
> *`ResourceName`*Ersetzen Sie durch den Namen der zu löschenden Ressource.
>
> ```
> kubectl delete ResourceType ResourceName -n $ACK_SYSTEM_NAMESPACE
> ```

Damit ist die Bereitstellung des neuen Workspace abgeschlossen. Im nächsten Abschnitt wird beschrieben, wie Sie Ihren Cluster so konfigurieren, dass Metriken an diesen Workspace gesendet werden.

Den Amazon-EKS-Cluster für das Schreiben im Workspace in Amazon Managed Service für Prometheus konfigurieren

In diesem Abschnitt wird beschrieben, wie Sie mithilfe des Befehls „Helm“ den in Ihrem Amazon-EKS-Cluster ausgeführten Prometheus-Service so konfigurieren, dass er Remote-Write-Metriken in den Workspace von Managed Service für Prometheus schreibt, den Sie im vorherigen Abschnitt erstellt haben.

Für dieses Verfahren benötigen Sie den Namen der IAM-Rolle, die Sie für die Erfassung von Metriken erstellt haben. Wenn Sie dies noch nicht getan haben, finden Sie weitere Informationen und Anweisungen unter Einrichten von Servicerollen für die Erfassung von Metriken aus Amazon-EKS-Clustern. Wenn Sie diese Anweisungen befolgen, wird die IAM-Rolle `amp-iamproxy-ingest-role` genannt.

So konfigurieren Sie Ihren Amazon-EKS-Cluster für Remote-Write

1. Verwenden Sie den folgenden Befehl, um `prometheusEndpoint` für Ihren Workspace zu erhalten. Ersetze es *`WORKSPACE-ID`* durch die Workspace-ID aus dem vorherigen Abschnitt.

```
aws amp describe-workspace --workspace-id WORKSPACE-ID
```

prometheusEndpunkt ist in den Rückgabeergebnissen enthalten und ist wie folgt formatiert:

```
https://aps-workspaces.us-west-2.amazonaws.com/workspaces/ws-a1b2c3d4-a123-b456-
c789-ac1234567890/
```

Speichern Sie diese URL zur Verwendung in den nächsten Schritten.

2. Erstellen Sie eine Datei mit folgendem Text und nennen Sie sie `prometheus-config.yaml`. Ersetze es *`account`* durch deine Konto-ID, *`workspaceURL/`* durch die URL, die du gerade gefunden hast, und *`region`* durch die AWS-Region für dein System passende URL.

```
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. Suchen Sie die Namen des Prometheus-Charts und -Namespaces sowie die Chart-Version mit dem folgenden Helm-Befehl.

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

Basierend auf den bisherigen Schritten sollten sowohl das Prometheus-Chart als auch der Namespace `prometheus` genannt werden, während es sich bei der Chart-Version um handeln `15.2.0` könnte.

4. Führen Sie den folgenden Befehl aus, und verwenden Sie dabei das *PrometheusChartNamePrometheusNamespace*, *PrometheusChartVersion* und, aus dem vorherigen Schritt.

```
helm upgrade PrometheusChartName prometheus-community/prometheus -
n PrometheusNamespace -f prometheus-config.yaml --version PrometheusChartVersion
```

   Nach einigen Minuten wird eine Nachricht angezeigt, dass die Aktualisierung erfolgreich war.

5. Überprüfen Sie optional, ob die Metriken erfolgreich gesendet wurden, indem Sie den Endpunkt von Amazon Managed Service für Prometheus über `awscurl` abfragen. *Region*Ersetzen Sie es durch AWS-Region die, die Sie verwenden, und *workspaceURL/* durch die URL, die Sie in Schritt 1 gefunden haben.

```
awscurl --service="aps" --region="Region" "workspaceURL/api/v1/query?
query=node_cpu_seconds_total"
```

Sie haben jetzt einen Workspace für Amazon Managed Service für Prometheus erstellt und von Ihrem Amazon-EKS-Cluster aus eine Verbindung zu ihm hergestellt, wobei YAML-Dateien als Konfiguration verwendet wurden. Diese Dateien, die als benutzerdefinierte Ressourcendefinitionen (CRDs) bezeichnet werden, befinden sich in Ihrem Amazon EKS-Cluster. Sie können den AWS Controller Controllers for Kubernetes verwenden, um all Ihre Amazon Managed Service for Prometheus-Ressourcen direkt vom Cluster aus zu verwalten.

Integration von CloudWatch Metriken mit Amazon Managed Service für Prometheus

Es kann hilfreich sein, alle Ihre Kennzahlen an einem Ort zu haben. Amazon Managed Service for Prometheus nimmt Amazon-Metriken nicht automatisch auf. CloudWatch Sie können jedoch Amazon Data Firehose verwenden und CloudWatch Metriken AWS Lambda an Amazon Managed Service for Prometheus übertragen.

In diesem Abschnitt wird beschrieben, wie Sie einen CloudWatch Amazon-Metrik-Stream instrumentieren und Amazon Data Firehose verwenden und AWS LambdaMetriken in Amazon Managed Service for Prometheus aufnehmen.

Sie richten mithilfe des AWS Cloud Development Kit (CDK) einen Stack ein, um einen Firehose Delivery Stream, einen Lambda und einen Amazon S3 S3-Bucket zu erstellen, um ein vollständiges Szenario zu demonstrieren.

Infrastruktur

Als Erstes müssen Sie die Infrastruktur für dieses Rezept einrichten.

CloudWatch Metrik-Streams ermöglichen die Weiterleitung der Streaming-Metrikdaten an einen HTTP-Endpunkt oder einen Amazon S3 S3-Bucket.

Die Einrichtung der Infrastruktur besteht aus 4 Schritten:

- Voraussetzungen konfigurieren
- Workspace von Amazon Managed Service für Prometheus erstellen
- Installieren von Abhängigkeiten
- Den Stack bereitstellen

Voraussetzungen

- Das AWS CLI ist in Ihrer Umgebung installiert und konfiguriert.
- Das AWS CDK -Typescript ist in Ihrer Umgebung installiert.
- Node.js und Go sind in Ihrer Umgebung installiert.
- Das Github-Repository (`CWMetricsStreamExporter`) für den Export von AWS CloudWatch Observability-Metriken wurde auf Ihren lokalen Computer geklont.

So erstellen Sie einen Workspace in Amazon Managed Service für Prometheus

1. Die Demo-App in diesem Rezept wird auf Amazon Managed Service für Prometheus ausgeführt. Erstellen Sie Ihren Workspace für Amazon Managed Service für Prometheus mithilfe des folgenden Befehls:

   ```
   aws amp create-workspace --alias prometheus-demo-recipe
   ```

2. Stellen Sie sicher, dass Ihr Workspace mit dem folgenden Befehl erstellt wurde:

   ```
   aws amp list-workspaces
   ```

Weitere Informationen zu Amazon Managed Service für Prometheus finden Sie im Benutzerhandbuch von Amazon Managed Service für Prometheus.

So installieren Sie Abhängigkeiten

1. Abhängigkeiten installieren

   Ändern Sie im Stammverzeichnis des Repositorys `aws-o11y-recipes` Ihr Verzeichnis zu `CWMetricStreamExporter` mit dem folgenden Befehl:

   ```
   cd sandbox/CWMetricStreamExporter
   ```

   Dies wird nun in Zukunft als das Stammverzeichnis des Repos betrachtet.

2. Wechseln Sie das Verzeichnis zu `/cdk` über den folgenden Befehl:

   ```
   cd cdk
   ```

3. Führen Sie den folgenden Befehl aus, um die CDK-Abhängigkeit zu installieren.

   ```
   npm install
   ```

4. Wechseln Sie zurück zum Stammverzeichnis des Repos und wechseln Sie dann das Verzeichnis zu `/lambda` mit dem folgenden Befehl:

   ```
   cd lambda
   ```

5. Sobald Sie sich im Ordner `/lambda` befinden, installieren Sie die Go-Abhängigkeiten mit:

   ```
   go get
   ```

   Alle Abhängigkeiten sind jetzt installiert.

So stellen Sie den Stack bereit

1. Öffnen Sie `config.yaml` im Stammverzeichnis des Repos und ändern Sie die Workspace-URL von Amazon Managed Service für Prometheus, indem Sie `{workspace}` durch die neu erstellte

Workspace-ID und die Region ersetzen, in der sich Ihr Workspace in Amazon Managed Service für Prometheus befindet.

Nutzen Sie dazu beispielsweise Folgendes:

```
AMP:
    remote_write_url: "https://aps-workspaces.us-east-2.amazonaws.com/workspaces/
{workspaceId}/api/v1/remote_write"
    region: us-east-2
```

Ändern Sie die Namen des Firehose-Lieferstreams und des Amazon S3 S3-Buckets nach Ihren Wünschen.

2. Um den AWS CDK und den Lambda-Code zu erstellen, führen Sie im Stammverzeichnis des Repos den folgenden Befehl aus:

```
npm run build
```

Dieser Build-Schritt stellt sicher, dass die Go Lambda-Binärdatei erstellt wird, und stellt das CDK für bereit. CloudFormation

3. Um die Bereitstellung abzuschließen, überprüfen und akzeptieren Sie die für den Stack erforderlichen IAM-Änderungen.
4. Optional können Sie überprüfen, ob der Stack erstellt wurde, indem Sie den folgenden Befehl ausführen.

```
aws cloudformation list-stacks
```

Ein Stack mit dem Namen `CDK Stack` wird in der Liste angezeigt.

Einen CloudWatch Amazon-Stream erstellen

Da Sie nun über eine Lambda-Funktion zur Verarbeitung der Metriken verfügen, können Sie den Metrik-Stream von Amazon CloudWatch erstellen.

Um einen CloudWatch Metrik-Stream zu erstellen

1. Gehen Sie zur CloudWatch Konsole unter https://console.aws.amazon.com/cloudwatch/#metric-streams:StreamsList und wählen Sie Metrik-Stream erstellen aus.

2. Wählen Sie die benötigten Metriken aus, entweder alle Metriken oder nur die aus ausgewählten Namespaces.
3. Wählen Sie unter `Configuration` die Option Vorhandenes Firehose auswählen, das Ihrem Konto gehört.
4. Sie werden den Firehose verwenden, der zuvor vom CDK erstellt wurde. Wählen Sie in der Dropdownliste Wählen Sie Ihren Kinesis-Daten-Firehose-Stream aus den zuvor erstellten Stream aus. Er wird einen Namen wie `CdkStack-KinesisFirehoseStream123456AB-sample1234` haben.
5. Ändern Sie das Ausgabeformat in JSON.
6. Geben Sie dem Metrik-Stream einen Namen, der für Sie von Bedeutung ist.
7. Wählen Sie Metrikstream erstellen aus.
8. (Optional) Um den Lambda-Funktionsaufruf zu überprüfen, navigieren Sie zur Lambda-Konsole und wählen Sie die Funktion `KinesisMessageHandler` aus. Wählen Sie die Registerkarte Monitor und die Unterregisterkarte Protokolle aus. Unter Letzte Aufrufe sollten Einträge der Lambda-Funktion angezeigt werden, die ausgelöst wird.

> **Note**
>
> Es kann bis zu 5 Minuten dauern, bis Aufrufe auf der Registerkarte Überwachen angezeigt werden.

Ihre Messwerte werden jetzt von Amazon CloudWatch an Amazon Managed Service for Prometheus gestreamt.

Bereinigen

Möglicherweise möchten Sie die Ressourcen bereinigen, die in diesem Beispiel verwendet wurden. In den folgenden Verfahren wird erläutert, wie das geht. Dadurch wird der von Ihnen erstellte Metrik-Stream gestoppt.

So bereinigen Sie Ressourcen

1. Löschen Sie zunächst den CloudFormation Stack mit den folgenden Befehlen:

```
cd cdk
cdk destroy
```

2. Workspace von Amazon Managed Service für Prometheus entfernen:

```
aws amp delete-workspace --workspace-id \
    `aws amp list-workspaces --alias prometheus-sample-app --query
 'workspaces[0].workspaceId' --output text`
```

3. Entfernen Sie abschließend den Amazon CloudWatch Metric Stream mithilfe der CloudWatch Amazon-Konsole.

Sicherheit in Amazon Managed Service für Prometheus

Cloud-Sicherheit hat höchste AWS Priorität. Als AWS Kunde profitieren Sie von Rechenzentren und Netzwerkarchitekturen, die darauf ausgelegt sind, die Anforderungen der sicherheitssensibelsten Unternehmen zu erfüllen.

Sicherheit ist eine gemeinsame AWS Verantwortung von Ihnen und Ihnen. Das Modell der geteilten Verantwortung beschreibt dies als Sicherheit der Cloud und Sicherheit in der Cloud:

- Sicherheit der Cloud — AWS ist verantwortlich für den Schutz der Infrastruktur, die AWS Dienste in der AWS Cloud ausführt. AWS bietet Ihnen auch Dienste, die Sie sicher nutzen können. Externe Prüfer testen und verifizieren regelmäßig die Wirksamkeit unserer Sicherheitsmaßnahmen im Rahmen der AWS . Weitere Informationen zu den Compliance-Programmen, die für Amazon Managed Service for Prometheus gelten, finden Sie unter AWS Services im Bereich nach Compliance-Programm AWS .
- Sicherheit in der Cloud — Ihre Verantwortung richtet sich nach dem AWS Service, den Sie nutzen. Sie sind auch für andere Faktoren verantwortlich, etwa für die Vertraulichkeit Ihrer Daten, für die Anforderungen Ihres Unternehmens und für die geltenden Gesetze und Vorschriften.

Diese Dokumentation zeigt Ihnen, wie Sie das Modell der übergreifenden Verantwortlichkeit bei der Verwendung von Amazon Managed Service für Prometheus einsetzen können. Die folgenden Themen zeigen Ihnen, wie Sie Amazon Managed Service für Prometheus konfigurieren, um Ihre Sicherheits- und Compliance-Ziele zu erreichen. Sie erfahren auch, wie Sie andere AWS Dienste nutzen können, die Ihnen helfen, Ihre Amazon Managed Service for Prometheus-Ressourcen zu überwachen und zu sichern.

Topics

- Datenschutz in Amazon Managed Service für Prometheus
- Identitäts- und Zugriffsverwaltung für Amazon Managed Service für Prometheus
- IAM-Berechtigungen und Richtlinien
- Compliance-Validierung für Amazon Managed Service für Prometheus
- Ausfallsicherheit in Amazon Managed Service für Prometheus
- Sicherheit der Infrastruktur in Amazon Managed Service für Prometheus
- Verwenden von serviceverknüpften Rollen für Amazon Managed Service für Prometheus

- Protokollieren von Amazon Managed Service for Prometheus API-Aufrufen mithilfe von AWS CloudTrail
- IAM-Rollen für Servicekonten einrichten
- Verwendung von Amazon Managed Service für Prometheus mit Schnittstellen-VPC-Endpunkten

Datenschutz in Amazon Managed Service für Prometheus

Das AWS Modell der gilt für den Datenschutz in Amazon Managed Service for Prometheus. Wie in diesem Modell beschrieben, AWS ist verantwortlich für den Schutz der globalen Infrastruktur, auf der alle Systeme laufen. AWS Cloud Sie sind dafür verantwortlich, die Kontrolle über Ihre in dieser Infrastruktur gehosteten Inhalte zu behalten. Sie sind auch für die Sicherheitskonfiguration und die Verwaltungsaufgaben für die von Ihnen verwendeten AWS-Services verantwortlich. Weitere Informationen zum Datenschutz finden Sie unter Häufig gestellte Fragen zum Datenschutz . Informationen zum Datenschutz in Europa finden Sie im General Data Protection Regulation (GDPR) Center.

Aus Datenschutzgründen empfehlen wir, dass Sie Ihre AWS-Konto Anmeldeinformationen schützen und einzelne Benutzer mit AWS IAM Identity Center oder AWS Identity and Access Management (IAM) einrichten. So erhält jeder Benutzer nur die Berechtigungen, die zum Durchführen seiner Aufgaben erforderlich sind. Außerdem empfehlen wir, die Daten mit folgenden Methoden schützen:

- Verwenden Sie für jedes Konto die Multi-Faktor-Authentifizierung (MFA).
- Wird verwendet SSL/TLS , um mit AWS Ressourcen zu kommunizieren. Wir benötigen TLS 1.2 und empfehlen TLS 1.3.
- Richten Sie die API und die Protokollierung von Benutzeraktivitäten mit ein AWS CloudTrail. Informationen zur Verwendung von CloudTrail Pfaden zur Erfassung von AWS Aktivitäten finden Sie unter Arbeiten mit CloudTrail Pfaden im AWS CloudTrail Benutzerhandbuch.
- Verwenden Sie AWS Verschlüsselungslösungen zusammen mit allen darin enthaltenen Standardsicherheitskontrollen AWS-Services.
- Verwenden Sie erweiterte verwaltete Sicherheitsservices wie Amazon Macie, die dabei helfen, in Amazon S3 gespeicherte persönliche Daten zu erkennen und zu schützen.
- Wenn Sie für den Zugriff AWS über eine Befehlszeilenschnittstelle oder eine API FIPS 140-3-validierte kryptografische Module benötigen, verwenden Sie einen FIPS-Endpunkt. Weitere Informationen über verfügbare FIPS-Endpunkte finden Sie unter Federal Information Processing Standard (FIPS) 140-3.

Wir empfehlen dringend, in Freitextfeldern, z. B. im Feld Name, keine vertraulichen oder sensiblen Informationen wie die E-Mail-Adressen Ihrer Kunden einzugeben. Dies gilt auch, wenn Sie mit Amazon Managed Service for Prometheus oder anderen AWS-Services über die Konsole AWS CLI, API oder AWS SDKs arbeiten. Alle Daten, die Sie in Tags oder Freitextfelder eingeben, die für Namen verwendet werden, können für Abrechnungs- oder Diagnoseprotokolle verwendet werden. Wenn Sie eine URL für einen externen Server bereitstellen, empfehlen wir dringend, keine Anmeldeinformationen zur Validierung Ihrer Anforderung an den betreffenden Server in die URL einzuschließen.

Themen

- Von Amazon Managed Service für Prometheus gesammelte Daten
- Verschlüsselung im Ruhezustand

Von Amazon Managed Service für Prometheus gesammelte Daten

Amazon Managed Service für Prometheus sammelt und speichert Betriebsmetriken, die Sie so konfigurieren, dass sie von Prometheus-Servern, die in Ihrem Konto laufen, an Amazon Managed Service für Prometheus gesendet werden. Die Daten umfassen Folgendes:

- Metrikwerte
- Metrische Bezeichnungen (oder beliebige Schlüssel-Wert-Paare), die bei der Identifizierung und Klassifizierung von Daten helfen
- Zeitstempel für Datenbeispielen

Eindeutige Tenant-IDs isolieren Daten von verschiedenen Kunden. Diese IDs schränken ein, auf welche Kundendaten zugegriffen werden kann. Kunden können Tenant-IDs nicht ändern.

Amazon Managed Service for Prometheus verschlüsselt die gespeicherten Daten mit AWS Key Management Service ()AWS KMS-Schlüsseln. Amazon Managed Service für Prometheus verwaltet diese Schlüssel.

> **Note**
>
> Amazon Managed Service for Prometheus unterstützt die Erstellung von kundenverwalteten Schlüsseln zur Verschlüsselung Ihrer Daten. Weitere Informationen zu den Schlüsseln, die Amazon Managed Service for Prometheus standardmäßig verwendet, und zur

Verwendung Ihrer eigenen kundenverwalteten Schlüssel finden Sie unter. Verschlüsselung im Ruhezustand

Daten werden während der Übertragung automatisch mit HTTPS verschlüsselt. Amazon Managed Service for Prometheus sichert Verbindungen zwischen Availability Zones innerhalb einer AWS Region mithilfe von HTTPS intern.

Verschlüsselung im Ruhezustand

Standardmäßig bietet Ihnen Amazon Managed Service for Prometheus automatisch Verschlüsselung im Ruhezustand, wobei AWS eigene Verschlüsselungsschlüssel verwendet werden.

- AWS eigene Schlüssel — Amazon Managed Service for Prometheus verwendet diese Schlüssel, um Daten, die in Ihren Workspace hochgeladen wurden, automatisch zu verschlüsseln. Sie können AWS eigene Schlüssel nicht einsehen, verwalten oder verwenden oder deren Verwendung überprüfen. Sie müssen jedoch keine Maßnahmen ergreifen oder Programme zum Schutz der Schlüssel ändern, die zur Verschlüsselung Ihrer Daten verwendet werden. Weitere Informationen finden Sie unter AWS -eigene Schlüssel im AWS Key Management Service -Entwicklerhandbuch.

Die Verschlüsselung von Daten im Ruhezustand trägt dazu bei, den betrieblichen Aufwand und die Komplexität zu reduzieren, die mit dem Schutz sensibler Kundendaten, z. B. personenbezogener Daten, verbunden sind. Sie können damit sichere Anwendungen erstellen, die strenge Verschlüsselungsvorschriften und gesetzliche Auflagen erfüllen.

Wenn Sie Ihren Workspace erstellen, können Sie sich alternativ dafür entscheiden, einen kundenverwalteten Schlüssel zu verwenden:

- Kundenverwaltete Schlüssel – Amazon Managed Service for Prometheus unterstützt die Verwendung eines symmetrischen, kundenverwalteten Schlüssels, den Sie erstellen, besitzen und verwalten, um die Daten in Ihrem Workspace zu verschlüsseln. Da Sie die volle Kontrolle über diese Verschlüsselung haben, können Sie Aufgaben wie die folgenden ausführen:
  - Festlegung und Pflege wichtiger Richtlinien
  - Festlegung und Aufrechterhaltung von IAM-Richtlinien und -Zuschüssen
  - Aktivieren und Deaktivieren wichtiger Richtlinien
  - Kryptographisches Material mit rotierendem Schlüssel
  - Hinzufügen von -Tags

- Erstellen von Schlüsselaliasen
- Schlüssel für das Löschen von Schlüsseln planen

Weitere Informationen finden Sie unter von Kunden verwaltete Schlüssel im AWS Key Management Service -Entwicklerhandbuch.

Wählen Sie sorgfältig aus, ob Sie vom Kunden verwaltete Schlüssel oder AWS eigene Schlüssel verwenden möchten. Workspaces, die mit kundenverwalteten Schlüsseln erstellt wurden, können später nicht mehr in die Verwendung AWS eigener Schlüssel umgewandelt werden (und umgekehrt).

> **Note**
>
> Amazon Managed Service for Prometheus aktiviert automatisch die Verschlüsselung im Ruhezustand mithilfe AWS eigener Schlüssel, um Ihre Daten kostenlos zu schützen. Für die Verwendung eines vom Kunden verwalteten Schlüssels fallen jedoch AWS KMS Gebühren an. Weitere Informationen zur Preisgestaltung finden Sie unter AWS Key Management Service – Preise.

Weitere Informationen zu AWS KMS finden Sie unter Was ist AWS Key Management Service?

> **Note**
>
> Workspaces, die mit vom Kunden verwalteten Schlüsseln erstellt wurden, können keine von AWS verwalteten Sammler für die Datenerfassung verwenden.

So verwendet Amazon Managed Service for Prometheus Zuschüsse in AWS KMS

Amazon Managed Service for Prometheus benötigt drei Erteilungen, um Ihren kundenverwalteten Schlüssel verwenden zu können.

Wenn Sie einen Amazon Managed Service for Prometheus Workspace erstellen, der mit einem vom Kunden verwalteten Schlüssel verschlüsselt ist, erstellt Amazon Managed Service for Prometheus die drei Grants in Ihrem Namen, indem es Anfragen an sendet. CreateGrant AWS KMS Grants in AWS KMS werden verwendet, um Amazon Managed Service for Prometheus Zugriff auf den KMS-Schlüssel in Ihrem Konto zu gewähren, auch wenn dieser nicht direkt in Ihrem Namen aufgerufen wird (z. B. beim Speichern von Metrikdaten, die aus einem Amazon EKS-Cluster gescrapt wurden).

Amazon Managed Service for Prometheus benötigt die Erteilung, um Ihren kundenverwalteten Schlüssel für die folgenden internen Vorgänge zu verwenden:

- Senden Sie DescribeKeyAnfragen an, um AWS KMS zu überprüfen, ob der symmetrische, vom Kunden verwaltete KMS-Schlüssel, den Sie bei der Erstellung eines Workspace angegeben haben, gültig ist.
- Senden Sie GenerateDataKeyAnfragen AWS KMS zur Generierung von Datenschlüsseln, die mit Ihrem vom Kunden verwalteten Schlüssel verschlüsselt sind.
- Senden Sie Entschlüsselungsanfragen an AWS KMS , um die verschlüsselten Datenschlüssel zu entschlüsseln, sodass sie zur Verschlüsselung Ihrer Daten verwendet werden können.

Amazon Managed Service for Prometheus gewährt dem AWS KMS Schlüssel drei Zuschüsse, sodass Amazon Managed Service for Prometheus den Schlüssel in Ihrem Namen verwenden kann. Sie können den Zugriff auf den Schlüssel entziehen, indem Sie die Schlüsselrichtlinie ändern, den Schlüssel deaktivieren oder die Erteilung widerrufen. Machen Sie sich mit den Konsequenzen dieser Aktionen vertraut, bevor Sie sie ausführen. Andernfalls kann es zu Datenverlusten in Ihrem Workspace kommen.

Wenn Sie den Zugriff auf eine der Erteilungen widerrufen, kann Amazon Managed Service for Prometheus mit dem kundenverwalteten Schlüssel weder auf die verschlüsselten Daten zugreifen noch neue an den Workspace gesendete Daten speichern, was sich nachteilig auf die Vorgänge auswirkt, die auf diese Daten angewiesen sind. Auf neue an den Workspace gesendete Daten kann nicht zugegriffen werden und sie können dauerhaft verloren gehen.

⚠ Warning

- Wenn Sie den Schlüssel deaktivieren oder den Zugriff auf Amazon Managed Service for Prometheus in der Schlüsselrichtlinie entfernen, ist der Zugriff auf die Workspace-Daten nicht mehr möglich. Auf neue Daten, die an den Workspace gesendet werden, kann nicht zugegriffen werden und sie können dauerhaft verloren gehen.

  Sie können Zugriff auf die Workspace-Daten erhalten und wieder neue Daten empfangen, indem Sie den Zugriff von Amazon Managed Service for Prometheus auf den Schlüssel wiederherstellen.
- Wenn Sie eine Erteilung widerrufen, kann sie nicht wiederhergestellt werden, und die Daten im Workspace gehen dauerhaft verloren.

Schritt 1: Erstellen eines kundenverwalteten Schlüssels

Sie können einen symmetrischen, vom Kunden verwalteten Schlüssel mithilfe der AWS-Managementkonsole APIs oder erstellen. AWS KMS Der Schlüssel muss sich nicht in demselben Konto wie der Workspace in Amazon Managed Service for Prometheus befinden, solange Sie wie unten beschrieben über die Richtlinie den erforderlichen Zugriff gewähren.

Einen symmetrischen kundenverwalteten Schlüssel erstellen

Folgen Sie den Schritten zum Erstellen eines symmetrischen kundenverwalteten Schlüssels im Entwicklerhandbuch zum AWS Key Management Service .

Schlüsselrichtlinie

Schlüsselrichtlinien steuern den Zugriff auf den vom Kunden verwalteten Schlüssel. Jeder vom Kunden verwaltete Schlüssel muss über genau eine Schlüsselrichtlinie verfügen, die aussagt, wer den Schlüssel wie verwenden kann. Wenn Sie Ihren kundenseitig verwalteten Schlüssel erstellen, können Sie eine Schlüsselrichtlinie angeben. Weitere Informationen finden Sie unter Verwalten des Zugriffs auf kundenverwaltete Schlüssel im Entwicklerhandbuch zum AWS Key Management Service .

In der Schlüsselrichtlinie müssen die folgenden API-Vorgänge zugelassen sein, um Ihren kundenverwalteten Schlüssel mit Ihren Workspaces in Amazon Managed Service for Prometheus zu verwenden:

- `kms:CreateGrant`: Fügt einem kundenverwalteten Schlüssel eine Erteilung hinzu. Gewährt Kontrollzugriff auf einen bestimmten KMS-Schlüssel, der den Zugriff auf die Erteilungsoperationen ermöglicht, die Amazon Managed Service for Prometheus benötigt. Weitere Informationen finden Sie unter Verwenden von Erteilungen im AWS Key Management Service -Entwicklerhandbuch.

  Dadurch kann Amazon Managed Service for Prometheus folgende Aktionen ausführen:

  - `GenerateDataKey` aufrufen, um einen verschlüsselten Datenschlüssel zu generieren und zu speichern, da der Datenschlüssel nicht sofort zum Verschlüsseln verwendet wird.
  - `Decrypt` aufrufen, um den gespeicherten verschlüsselten Datenschlüssel für den Zugriff auf verschlüsselte Daten zu verwenden.
- `kms:DescribeKey`: Stellt Details zu kundenverwalteten Schlüsseln bereit, damit der Amazon Managed Service for Prometheus den Schlüssel validieren kann.

Im Folgenden sind Beispiele für Richtlinienanweisungen aufgeführt, die Sie für den Amazon Managed Service for Prometheus hinzufügen können:

```
 "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>
 ]
```

- Weitere Informationen zum Festlegen von Berechtigungen in einer Richtlinie finden Sie im AWS Key Management Service -Entwicklerhandbuch.

- Weitere Informationen zur Fehlerbehebung beim Schlüsselzugriff finden Sie im AWS Key Management Service -Entwicklerhandbuch.

Schritt 2: Angeben eines vom Kunden verwalteten Schlüssels für Amazon Managed Service for Prometheus

Wenn Sie einen Workspace erstellen, können Sie den kundenverwalteten Schlüssel angeben, indem Sie einen KMS-Schlüssel-ARN eingeben, den Amazon Managed Service for Prometheus verwendet, um die im Workspace gespeicherten Daten zu verschlüsseln.

Schritt 3: Zugreifen auf Daten von anderen Diensten wie Amazon Managed Grafana

Dieser Schritt ist optional — er ist nur erforderlich, wenn Sie von einem anderen Service aus auf Ihre Amazon Managed Service for Prometheus-Daten zugreifen müssen.

Auf Ihre verschlüsselten Daten kann von anderen Diensten nicht zugegriffen werden, es sei denn, diese haben ebenfalls Zugriff auf den AWS KMS Schlüssel. Wenn Sie beispielsweise Amazon Managed Grafana verwenden möchten, um ein Dashboard oder eine Warnung für Ihre Daten zu erstellen, müssen Sie Amazon Managed Grafana Zugriff auf den Schlüssel gewähren.

Um Amazon Managed Grafana Zugriff auf Ihren vom Kunden verwalteten Schlüssel zu gewähren

1. Wählen Sie in Ihrer Liste der Amazon Managed Grafana-Workspaces den Namen für den Workspace aus, für den Sie Zugriff auf Amazon Managed Service for Prometheus haben möchten. Dies zeigt Ihnen zusammenfassende Informationen zu Ihrem Amazon Managed Grafana-Arbeitsbereich.
2. Notieren Sie sich den Namen der IAM-Rolle, die von Ihrem Workspace verwendet wird. Der Name hat das Format`AmazonGrafanaServiceRole-<unique-id>`. Die Konsole zeigt Ihnen den vollständigen ARN für die Rolle. Sie werden diesen Namen in einem späteren Schritt in der AWS KMS Konsole angeben.
3. Wählen Sie in Ihrer Liste „Vom AWS KMS Kunden verwaltete Schlüssel“ den vom Kunden verwalteten Schlüssel aus, den Sie bei der Erstellung Ihres Amazon Managed Service for Prometheus-Workspace verwendet haben. Dadurch wird die Seite mit den Details zur Schlüsselkonfiguration geöffnet.
4. Wählen Sie neben Hauptbenutzer die Schaltfläche Hinzufügen aus.
5. Wählen Sie aus der Namensliste die Amazon Managed Grafana-IAM-Rolle aus, die Sie oben notiert haben. Um die Suche zu erleichtern, können Sie auch nach dem Namen suchen.

6. Wählen Sie Hinzufügen, um die IAM-Rolle zur Liste der Hauptbenutzer hinzuzufügen.

Ihr Amazon Managed Grafana-Arbeitsbereich kann jetzt auf die Daten in Ihrem Amazon Managed Service for Prometheus-Arbeitsbereich zugreifen. Sie können den Hauptbenutzern weitere Benutzer oder Rollen hinzufügen, damit andere Dienste auf Ihren Workspace zugreifen können.

Verschlüsselungskontext im Amazon Managed Service for Prometheus

Ein Verschlüsselungskontext ist ein optionaler Satz von Schlüssel-Wert-Paaren, die zusätzliche kontextbezogene Informationen zu den Daten enthalten.

AWS KMS verwendet den Verschlüsselungskontext als zusätzliche authentifizierte Daten, um die authentifizierte Verschlüsselung zu unterstützen. Wenn Sie einen Verschlüsselungskontext in eine Anforderung zum Verschlüsseln von Daten einbeziehen, wird der Verschlüsselungskontext AWS KMS an die verschlüsselten Daten gebunden. Zur Entschlüsselung von Daten müssen Sie denselben Verschlüsselungskontext in der Anfrage übergeben.

Verschlüsselungskontext im Amazon Managed Service for Prometheus

Amazon Managed Service for Prometheus verwendet bei allen AWS KMS kryptografischen Vorgängen denselben Verschlüsselungskontext, wobei der Schlüssel `aws:amp:arn` und der Wert der Amazon-Ressourcenname (ARN) des Workspace ist.

Example

```
"encryptionContext": {
    "aws:aps:arn": "arn:aws:aps:us-west-2:111122223333:workspace/ws-sample-1234-
abcd-56ef-7890abcd12ef"
}
```

Verwenden des Verschlüsselungskontexts für die Überwachung

Wenn Sie einen symmetrischen, vom Kunden verwalteten Schlüssel verwenden, um Ihre Workspace-Daten zu verschlüsseln, können Sie den Verschlüsselungskontext auch in Prüfaufzeichnungen und Protokollen verwenden, um zu ermitteln, wie der kundenverwaltete Schlüssel verwendet wird. Der Verschlüsselungskontext erscheint auch in Protokollen, die von Amazon CloudWatch Logs generiert wurden AWS CloudTrail.

Verwendung des Verschlüsselungskontextes zur Steuerung des Zugriffs auf den vom Kunden verwalteten Schlüssel

Sie können den Verschlüsselungskontext in Schlüsselrichtlinien und IAM-Richtlinien als `conditions` verwenden, um den Zugriff auf Ihren symmetrischen, kundenverwalteten Schlüssel zu kontrollieren. Sie können Verschlüsselungskontext-Einschränkungen auch in einer Genehmigung verwenden.

Amazon Managed Service for Prometheus verwendet eine Einschränkung des Verschlüsselungskontextes bei Erteilungen, um den Zugriff auf den kundenverwalteten Schlüssel in Ihrem Konto oder Ihrer Region zu steuern. Eine Genehmigungseinschränkung erfordert, dass durch die Genehmigung ermöglichte Vorgänge den angegebenen Verschlüsselungskontext verwenden.

Example

Im Folgenden finden Sie Beispiele für Schlüsselrichtlinienanweisungen zur Gewährung des Zugriffs auf einen kundenverwalteten Schlüssel für einen bestimmten Verschlüsselungskontext. Die Bedingung in dieser Richtlinienanweisung setzt voraus, dass die Genehmigungen eine Einschränkung des Verschlüsselungskontextes haben, die den Verschlüsselungskontext spezifiziert.

```
{
    "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"
          }
     }
}
```

Überwachen Ihrer Verschlüsselungsschlüssel für Amazon Managed Service for Prometheus

Wenn Sie einen vom AWS KMS Kunden verwalteten Schlüssel mit Ihren Amazon Managed Service for Prometheus Workspaces verwenden, können Sie AWS CloudTrailoder Amazon CloudWatch Logs verwenden, um Anfragen nachzuverfolgen, an die Amazon Managed Service for Prometheus sendet. AWS KMS

Die folgenden Beispiele sind AWS CloudTrail Ereignisse für`CreateGrant`,`GenerateDataKey`, und `DescribeKey` zur Überwachung von KMS-Vorgängen`Decrypt`, die von Amazon Managed Service for Prometheus aufgerufen werden, um auf Daten zuzugreifen, die mit Ihrem vom Kunden verwalteten Schlüssel verschlüsselt wurden:

CreateGrant

Wenn Sie einen vom AWS KMS Kunden verwalteten Schlüssel verwenden, um Ihren Workspace zu verschlüsseln, sendet Amazon Managed Service for Prometheus in Ihrem Namen drei `CreateGrant` Anfragen, um auf den von Ihnen angegebenen KMS-Schlüssel zuzugreifen. Die von Amazon Managed Service for Prometheus erstellten Erteilungen beziehen sich nur auf die Ressource, die dem kundenverwalteten AWS KMS -Schlüssel zugeordnet ist.

Das folgende Beispielereignis veranschaulicht eine `CreateGrant`-Operation:

```
{
    "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

Wenn Sie einen vom AWS KMS Kunden verwalteten Schlüssel für Ihren Workspace aktivieren, erstellt Amazon Managed Service for Prometheus einen eindeutigen Schlüssel. Es sendet eine `GenerateDataKey` Anfrage an AWS KMS , in der der vom AWS KMS Kunden verwaltete Schlüssel für die Ressource angegeben wird.

Das folgende Beispielereignis zeichnet den Vorgang `GenerateDataKey` auf:

```
{
    "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

Wenn eine Abfrage in einem verschlüsselten Workspace generiert wird, ruft Amazon Managed Service for Prometheus die `Decrypt`-Operation auf, um den gespeicherten verschlüsselten Datenschlüssel für den Zugriff auf die verschlüsselten Daten zu verwenden.

Das folgende Beispielereignis zeichnet den Vorgang `Decrypt` auf:

```
{
    "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 überprüft anhand der `DescribeKey`-Operation, ob der kundenverwaltete AWS KMS -Schlüssel, der mit Ihrem Workspace verknüpft ist, in dem Konto und in der Region vorhanden ist.

Das folgende Beispielereignis zeichnet den Vorgang `DescribeKey` auf:

```
{
    "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"
}
```

Weitere Informationen

Die folgenden Ressourcen enthalten weitere Informationen zur Datenverschlüsselung im Ruhezustand:

- Weitere Informationen zu grundlegenden AWS Key Management Service -Konzepten finden Sie im AWS Key Management Service -Entwicklerhandbuch.
- Weitere Informationen zu bewährten Sicherheitsmethoden für AWS Key Management Service finden Sie im AWS Key Management Service Entwicklerhandbuch.

Identitäts- und Zugriffsverwaltung für Amazon Managed Service für Prometheus

AWS Identity and Access Management (IAM) hilft einem Administrator AWS-Service , den Zugriff auf AWS Ressourcen sicher zu kontrollieren. IAM-Administratoren steuern, wer authentifiziert (angemeldet) und autorisiert (im Besitz von Berechtigungen) ist, die Ressourcen von Amazon Managed Service für Prometheus zu nutzen. IAM ist ein Programm AWS-Service , das Sie ohne zusätzliche Kosten nutzen können.

Themen

- Zielgruppe
- Authentifizierung mit Identitäten
- Verwalten des Zugriffs mit Richtlinien
- So funktioniert Amazon Managed Service für Prometheus mit IAM
- Beispiele für identitätsbasierte Richtlinien für Amazon Managed Service für Prometheus
- Problembehebung von Identität und Zugriff auf Amazon Managed Service für Prometheus

Zielgruppe

Wie Sie AWS Identity and Access Management (IAM) verwenden, hängt von Ihrer Rolle ab:

- Servicebenutzer – Fordern Sie von Ihrem Administrator Berechtigungen an, wenn Sie nicht auf Features zugreifen können (siehe Problembehebung von Identität und Zugriff auf Amazon Managed Service für Prometheus).
- Serviceadministrator – Bestimmen Sie den Benutzerzugriff und stellen Sie Berechtigungsanfragen (siehe So funktioniert Amazon Managed Service für Prometheus mit IAM).
- IAM-Administrator – Schreiben Sie Richtlinien zur Zugriffsverwaltung (siehe Beispiele für identitätsbasierte Richtlinien für Amazon Managed Service für Prometheus).

Authentifizierung mit Identitäten

Authentifizierung ist die Art und Weise, wie Sie sich AWS mit Ihren Identitätsdaten anmelden. Sie müssen sich als IAM-Benutzer authentifizieren oder eine IAM-Rolle annehmen. Root-Benutzer des AWS-Kontos

Sie können sich als föderierte Identität anmelden, indem Sie Anmeldeinformationen aus einer Identitätsquelle wie AWS IAM Identity Center (IAM Identity Center), Single Sign-On-Authentifizierung oder Anmeldeinformationen verwenden. Google/Facebook Weitere Informationen zum Anmelden

finden Sie unter So melden Sie sich bei Ihrem AWS-Konto an im Benutzerhandbuch für AWS-Anmeldung .

AWS Bietet für den programmatischen Zugriff ein SDK und eine CLI zum kryptografischen Signieren von Anfragen. Weitere Informationen finden Sie unter AWS Signature Version 4 for API requests im IAM-Benutzerhandbuch.

AWS-Konto Root-Benutzer

Wenn Sie einen erstellen AWS-Konto, beginnen Sie mit einer Anmeldeidentität, dem sogenannten AWS-Konto Root-Benutzer, der vollständigen Zugriff auf alle AWS-Services Ressourcen hat. Wir raten ausdrücklich davon ab, den Root-Benutzer für Alltagsaufgaben zu verwenden. Eine Liste der Aufgaben, für die Sie sich als Root-Benutzer anmelden müssen, finden Sie unter Tasks that require root user credentials im IAM-Benutzerhandbuch.

Verbundidentität

Es hat sich bewährt, dass menschliche Benutzer für den Zugriff AWS-Services mithilfe temporärer Anmeldeinformationen einen Verbund mit einem Identitätsanbieter verwenden müssen.

Eine föderierte Identität ist ein Benutzer aus Ihrem Unternehmensverzeichnis, Ihrem Directory Service Web-Identitätsanbieter oder der AWS-Services mithilfe von Anmeldeinformationen aus einer Identitätsquelle zugreift. Verbundene Identitäten übernehmen Rollen, die temporäre Anmeldeinformationen bereitstellen.

Für die zentrale Zugriffsverwaltung empfehlen wir AWS IAM Identity Center. Weitere Informationen finden Sie unter Was ist IAM Identity Center? im AWS IAM Identity Center -Benutzerhandbuch.

IAM-Benutzer und -Gruppen

Ein IAM-Benutzer ist eine Identität mit bestimmten Berechtigungen für eine einzelne Person oder Anwendung. Wir empfehlen die Verwendung temporärer Anmeldeinformationen anstelle von IAM-Benutzern mit langfristigen Anmeldeinformationen. Weitere Informationen finden Sie im IAM-Benutzerhandbuch unter Erfordern, dass menschliche Benutzer den Verbund mit einem Identitätsanbieter verwenden müssen, um AWS mithilfe temporärer Anmeldeinformationen darauf zugreifen zu können.

Eine IAM-Gruppe spezifiziert eine Sammlung von IAM-Benutzern und erleichtert die Verwaltung von Berechtigungen für große Gruppen von Benutzern. Weitere Informationen finden Sie unter Anwendungsfälle für IAM-Benutzer im IAM-Benutzerhandbuch.

IAM-Rollen

Eine IAM-Rolle ist eine Identität mit spezifischen Berechtigungen, die temporäre Anmeldeinformationen bereitstellt. Sie können eine Rolle übernehmen, indem Sie von einer Benutzer- zu einer IAM-Rolle (Konsole) wechseln oder indem Sie eine AWS Oder-API-Operation AWS CLI aufrufen. Weitere Informationen finden Sie unter Methoden, um eine Rolle zu übernehmen im IAM-Benutzerhandbuch.

IAM-Rollen sind nützlich für den Verbundbenutzer-Zugriff, temporäre IAM-Benutzerberechtigungen, kontoübergreifenden Zugriff, serviceübergreifenden Zugriff und Anwendungen, die auf Amazon EC2 laufen. Weitere Informationen finden Sie unter Kontoübergreifender Ressourcenzugriff in IAM im IAM-Benutzerhandbuch.

Verwalten des Zugriffs mit Richtlinien

Sie kontrollieren den Zugriff, AWS indem Sie Richtlinien erstellen und diese an AWS Identitäten oder Ressourcen anhängen. Eine Richtlinie definiert Berechtigungen, wenn sie mit einer Identität oder Ressource verknüpft sind. AWS bewertet diese Richtlinien, wenn ein Principal eine Anfrage stellt. Die meisten Richtlinien werden AWS als JSON-Dokumente gespeichert. Weitere Informationen zu JSON-Richtliniendokumenten finden Sie unter Übersicht über JSON-Richtlinien im IAM-Benutzerhandbuch.

Mit Hilfe von Richtlinien legen Administratoren fest, wer Zugriff auf was hat, indem sie definieren, welches Prinzipal welche Aktionen auf welchen Ressourcenund unter welchen Bedingungendurchführen darf.

Standardmäßig haben Benutzer, Gruppen und Rollen keine Berechtigungen. Ein IAM-Administrator erstellt IAM-Richtlinien und fügt sie zu Rollen hinzu, die die Benutzer dann übernehmen können. IAM-Richtlinien definieren Berechtigungen unabhängig von der Methode, die zur Ausführung der Operation verwendet wird.

Identitätsbasierte Richtlinien

Identitätsbasierte Richtlinien sind JSON-Berechtigungsrichtliniendokumente, die Sie einer Identität (Benutzer, Gruppe oder Rolle) anfügen können. Diese Richtlinien steuern, welche Aktionen Identitäten für welche Ressourcen und unter welchen Bedingungen ausführen können. Informationen zum Erstellen identitätsbasierter Richtlinien finden Sie unter Definieren benutzerdefinierter IAM-Berechtigungen mit vom Kunden verwalteten Richtlinien im IAM-Benutzerhandbuch.

Identitätsbasierte Richtlinien können Inline-Richtlinien (direkt in eine einzelne Identität eingebettet) oder verwaltete Richtlinien (eigenständige Richtlinien, die mit mehreren Identitäten verbunden sind)

sein. Informationen dazu, wie Sie zwischen verwalteten und Inline-Richtlinien wählen, finden Sie unter Choose between managed policies and inline policies im IAM-Benutzerhandbuch.

Ressourcenbasierte Richtlinien

Ressourcenbasierte Richtlinien sind JSON-Richtliniendokumente, die Sie an eine Ressource anfügen. Beispiele hierfür sind Vertrauensrichtlinien für IAM-Rollen und Amazon S3-Bucket-Richtlinien. In Services, die ressourcenbasierte Richtlinien unterstützen, können Service-Administratoren sie verwenden, um den Zugriff auf eine bestimmte Ressource zu steuern. Sie müssen in einer ressourcenbasierten Richtlinie einen Prinzipal angeben.

Ressourcenbasierte Richtlinien sind Richtlinien innerhalb dieses Diensts. Sie können AWS verwaltete Richtlinien von IAM nicht in einer ressourcenbasierten Richtlinie verwenden.

Weitere Richtlinientypen

AWS unterstützt zusätzliche Richtlinientypen, mit denen die maximalen Berechtigungen festgelegt werden können, die durch gängigere Richtlinientypen gewährt werden:

- Berechtigungsgrenzen – Eine Berechtigungsgrenze legt die maximalen Berechtigungen fest, die eine identitätsbasierte Richtlinie einer IAM-Entität erteilen kann. Weitere Informationen finden Sie unter Berechtigungsgrenzen für IAM-Entitäten im -IAM-Benutzerhandbuch.
- Richtlinien zur Dienstkontrolle (SCPs) — Geben Sie die maximalen Berechtigungen für eine Organisation oder Organisationseinheit in an AWS Organizations. Weitere Informationen finden Sie unter Service-Kontrollrichtlinien im AWS Organizations -Benutzerhandbuch.
- Richtlinien zur Ressourcenkontrolle (RCPs) — Legen Sie die maximal verfügbaren Berechtigungen für Ressourcen in Ihren Konten fest. Weitere Informationen finden Sie im AWS Organizations Benutzerhandbuch unter Richtlinien zur Ressourcenkontrolle (RCPs).
- Sitzungsrichtlinien – Sitzungsrichtlinien sind erweiterte Richtlinien, die als Parameter übergeben werden, wenn Sie eine temporäre Sitzung für eine Rolle oder einen Verbundbenutzer erstellen. Weitere Informationen finden Sie unter Sitzungsrichtlinien im IAM-Benutzerhandbuch.

Mehrere Richtlinientypen

Wenn für eine Anfrage mehrere Arten von Richtlinien gelten, sind die sich daraus ergebenden Berechtigungen schwieriger zu verstehen. Informationen darüber, wie AWS bestimmt wird, ob eine Anfrage zulässig ist, wenn mehrere Richtlinientypen betroffen sind, finden Sie unter Bewertungslogik für Richtlinien im IAM-Benutzerhandbuch.

So funktioniert Amazon Managed Service für Prometheus mit IAM

Bevor Sie IAM zum Verwalten des Zugriffs auf Amazon Managed Service für Prometheus verwenden, erfahren Sie, welche IAM-Funktionen Sie mit Amazon Managed Service für Prometheus verwenden können.

IAM-Features, die mit Amazon Managed Service für Prometheus verwendet werden können

IAM-Feature	Support für Amazon Managed Service für Prometheus
Identitätsbasierte Richtlinien	Ja
Ressourcenbasierte Richtlinien	Ja
Richtlinienaktionen	Ja
Richtlinienressourcen	Ja
Bedingungsschlüssel für die Richtlinie	Nein
ACLs	Nein
ABAC (Tags in Richtlinien)	Ja
Temporäre Anmeldeinformationen	Ja
Forward Access Sessions (FAS)	Nein
Servicerollen	Nein
Serviceverknüpfte Rollen	Ja

Einen allgemeinen Überblick darüber, wie Amazon Managed Service for Prometheus und andere AWS Services mit den meisten IAM-Funktionen funktionieren, finden Sie im AWS IAM-Benutzerhandbuch unter Services, die mit IAM funktionieren.

Identitätsbasierte Richtlinien für Amazon Managed Service für Prometheus

Unterstützt Richtlinien auf Identitätsbasis: Ja

Identitätsbasierte Richtlinien sind JSON-Berechtigungsrichtliniendokumente, die Sie einer Identität anfügen können, wie z. B. IAM-Benutzern, -Benutzergruppen oder -Rollen. Diese Richtlinien steuern, welche Aktionen die Benutzer und Rollen für welche Ressourcen und unter welchen Bedingungen ausführen können. Informationen zum Erstellen identitätsbasierter Richtlinien finden Sie unter Definieren benutzerdefinierter IAM-Berechtigungen mit vom Kunden verwalteten Richtlinien im IAM-Benutzerhandbuch.

Mit identitätsbasierten IAM-Richtlinien können Sie angeben, welche Aktionen und Ressourcen zugelassen oder abgelehnt werden. Darüber hinaus können Sie die Bedingungen festlegen, unter denen Aktionen zugelassen oder abgelehnt werden. Informationen zu sämtlichen Elementen, die Sie in einer JSON-Richtlinie verwenden, finden Sie in der IAM-Referenz für JSON-Richtlinienelemente im IAM-Benutzerhandbuch.

Beispiele für identitätsbasierte Richtlinien für Amazon Managed Service für Prometheus

Beispiele für identitätsbasierte Richtlinien für Amazon Managed Service für Prometheus finden Sie unter Beispiele für identitätsbasierte Richtlinien für Amazon Managed Service für Prometheus.

Ressourcenbasierte Richtlinien in Amazon Managed Service für Prometheus

Unterstützt ressourcenbasierte Richtlinien: Ja

Ressourcenbasierte Richtlinien sind JSON-Richtliniendokumente, die Sie an eine Ressource anfügen. Beispiele für ressourcenbasierte Richtlinien sind IAM-Rollen-Vertrauensrichtlinien und Amazon-S3-Bucket-Richtlinien. In Services, die ressourcenbasierte Richtlinien unterstützen, können Service-Administratoren sie verwenden, um den Zugriff auf eine bestimmte Ressource zu steuern. Für die Ressource, an welche die Richtlinie angehängt ist, legt die Richtlinie fest, welche Aktionen ein bestimmter Prinzipal unter welchen Bedingungen für diese Ressource ausführen kann. Sie müssen in einer ressourcenbasierten Richtlinie einen Prinzipal angeben. Zu den Prinzipalen können Konten, Benutzer, Rollen, Verbundbenutzer oder gehören. AWS-Services

Um kontoübergreifenden Zugriff zu ermöglichen, können Sie ein gesamtes Konto oder IAM-Entitäten in einem anderen Konto als Prinzipal in einer ressourcenbasierten Richtlinie angeben. Weitere Informationen finden Sie unter Kontoübergreifender Ressourcenzugriff in IAM im IAM-Benutzerhandbuch.

Richtlinienaktionen für Amazon Managed Service für Prometheus

Unterstützt Richtlinienaktionen: Ja

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer Zugriff auf was hat. Das heißt, welcher Prinzipal Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen kann.

Das Element `Action` einer JSON-Richtlinie beschreibt die Aktionen, mit denen Sie den Zugriff in einer Richtlinie zulassen oder verweigern können. Nehmen Sie Aktionen in eine Richtlinie auf, um Berechtigungen zur Ausführung des zugehörigen Vorgangs zu erteilen.

Eine Liste der Aktionen in Amazon Managed Service für Prometheus finden Sie unter Von Amazon Managed Service für Prometheus definierte Aktionen in der Referenz für die Service-Autorisierung.

Richtlinienaktionen in Amazon Managed Service für Prometheus verwenden das folgende Präfix vor der Aktion:

```
aps
```

Um mehrere Aktionen in einer einzigen Anweisung anzugeben, trennen Sie sie mit Kommata:

```
"Action": [
      "aps:action1",
      "aps:action2"
         ]
```

Beispiele für identitätsbasierte Richtlinien für Amazon Managed Service für Prometheus finden Sie unter Beispiele für identitätsbasierte Richtlinien für Amazon Managed Service für Prometheus.

Richtlinienressourcen für Amazon Managed Service für Prometheus

Unterstützt Richtlinienressourcen: Ja

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer Zugriff auf was hat. Das heißt, welcher Prinzipal Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen kann.

Das JSON-Richtlinienelement `Resource` gibt die Objekte an, auf welche die Aktion angewendet wird. Als Best Practice geben Sie eine Ressource mit dem zugehörigen Amazon-Ressourcennamen (ARN) an. Verwenden Sie für Aktionen, die keine Berechtigungen auf Ressourcenebene unterstützen, einen Platzhalter (*), um anzugeben, dass die Anweisung für alle Ressourcen gilt.

```
"Resource": "*"
```

Eine Liste der Ressourcentypen und der zugehörigen Amazon Managed Service for Prometheus finden Sie unter Von Amazon Managed Service for Prometheus definierte Ressourcen in der Service Authorization Reference. ARNs Informationen zu den Aktionen, mit denen Sie den ARN einzelner Ressourcen angeben können, finden Sie unter Von Amazon Managed Service für Prometheus definierte Aktionen.

Beispiele für identitätsbasierte Richtlinien für Amazon Managed Service für Prometheus finden Sie unter Beispiele für identitätsbasierte Richtlinien für Amazon Managed Service für Prometheus.

Richtlinienbedingungsschlüssel für Amazon Managed Service für Prometheus

Unterstützt servicespezifische Richtlinienbedingungsschlüssel: Nein

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer Zugriff auf was hat. Das heißt, welcher Prinzipal Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen kann.

Das Element `Condition` gibt an, wann Anweisungen auf der Grundlage definierter Kriterien ausgeführt werden. Sie können bedingte Ausdrücke erstellen, die Bedingungsoperatoren verwenden, z. B. ist gleich oder kleiner als, damit die Bedingung in der Richtlinie mit Werten in der Anforderung übereinstimmt. Eine Übersicht aller AWS globalen Bedingungsschlüssel finden Sie unter Kontextschlüssel für AWS globale Bedingungen im IAM-Benutzerhandbuch.

Eine Liste der Bedingungsschlüsseln von Amazon Managed Service für Prometheus finden Sie unter Von Amazon Managed Service für Prometheus definierte Bedingungsschlüssel in der Referenz für die Service-Autorisierung. Informationen dazu, mit welchen Aktionen und Ressourcen Sie einen Bedingungsschlüssel verwenden können, finden Sie unter Von Amazon Managed Service für Prometheus definierte Aktionen.

Beispiele für identitätsbasierte Richtlinien für Amazon Managed Service für Prometheus finden Sie unter Beispiele für identitätsbasierte Richtlinien für Amazon Managed Service für Prometheus.

Zugriffskontrolllisten (ACLs) in Amazon Managed Service für Prometheus

Unterstützt ACLs: Nein

Zugriffskontrolllisten (ACLs) steuern, welche Principals (Kontomitglieder, Benutzer oder Rollen) über Zugriffsberechtigungen für eine Ressource verfügen. ACLs ähneln ressourcenbasierten Richtlinien, verwenden jedoch nicht das JSON-Richtliniendokumentformat.

Attributbasierte Zugriffssteuerung (ABAC) mit Amazon Managed Service für Prometheus

Unterstützt ABAC (Tags in Richtlinien): Ja

Die attributbasierte Zugriffskontrolle (ABAC) ist eine Autorisierungsstrategie, bei der Berechtigungen basierend auf Attributen, auch als Tags bezeichnet, definiert werden. Sie können Tags an IAM-Entitäten und AWS -Ressourcen anhängen und dann ABAC-Richtlinien so entwerfen, dass Operationen möglich sind, wenn das Tag des Prinzipals mit dem Tag auf der Ressource übereinstimmt.

Um den Zugriff auf der Grundlage von Tags zu steuern, geben Sie im Bedingungselement einer Richtlinie Tag-Informationen an, indem Sie die Schlüssel `aws:ResourceTag/`*`key-name`*, `aws:RequestTag/`*`key-name`*, oder Bedingung `aws:TagKeys` verwenden.

Wenn ein Service alle drei Bedingungsschlüssel für jeden Ressourcentyp unterstützt, lautet der Wert für den Service Ja. Wenn ein Service alle drei Bedingungsschlüssel für nur einige Ressourcentypen unterstützt, lautet der Wert Teilweise.

Weitere Informationen zu ABAC finden Sie unter Definieren von Berechtigungen mit ABAC-Autorisierung im IAM-Benutzerhandbuch. Um ein Tutorial mit Schritten zur Einstellung von ABAC anzuzeigen, siehe Attributbasierte Zugriffskontrolle (ABAC) verwenden im IAM-Benutzerhandbuch.

Verwenden temporärer Anmeldeinformationen mit Amazon Managed Service für Prometheus

Unterstützt temporäre Anmeldeinformationen: Ja

Temporäre Anmeldeinformationen ermöglichen den kurzfristigen Zugriff auf AWS Ressourcen und werden automatisch erstellt, wenn Sie einen Verbund verwenden oder die Rollen wechseln. AWS empfiehlt, temporäre Anmeldeinformationen dynamisch zu generieren, anstatt langfristige Zugriffsschlüssel zu verwenden. Weitere Informationen finden Sie unter Temporäre Anmeldeinformationen in IAM und AWS-Services , die mit IAM funktionieren im IAM-Benutzerhandbuch.

Forward Access Sessions für Amazon Managed Service for Prometheus

Unterstützt Forward Access Sessions (FAS): Nein

Forward Access Sessions (FAS) verwenden die Berechtigungen des Prinzipals, der einen aufruft AWS-Service, in Kombination mit der Anfrage, Anfragen AWS-Service an nachgeschaltete Dienste zu stellen. Einzelheiten zu den Richtlinien für FAS-Anfragen finden Sie unter Zugriffssitzungen weiterleiten.

Servicerollen für Amazon Managed Service für Prometheus

Unterstützt Servicerollen: Nein

Eine Servicerolle ist eine IAM-Rolle, die ein Service annimmt, um Aktionen in Ihrem Namen auszuführen. Ein IAM-Administrator kann eine Servicerolle innerhalb von IAM erstellen, ändern und löschen. Weitere Informationen finden Sie unter Erstellen einer Rolle zum Delegieren von Berechtigungen an einen AWS-Service im IAM-Benutzerhandbuch.

> **Warning**
>
> Das Ändern der Berechtigungen für eine Servicerolle könnte die Funktionalität von Amazon Managed Service für Prometheus beeinträchtigen. Bearbeiten Sie Servicerollen nur, wenn Amazon Managed Service für Prometheus dazu Anleitungen gibt.

Serviceverknüpfte Rollen für Amazon Managed Service für Prometheus

Unterstützt serviceverknüpfte Rollen: Ja

Eine serviceverknüpfte Rolle ist eine Art von Servicerolle, die mit einer verknüpft ist. AWS-Service Der Service kann die Rolle übernehmen, um eine Aktion in Ihrem Namen auszuführen. Dienstbezogene Rollen werden in Ihrem Dienst angezeigt AWS-Konto und gehören dem Dienst. Ein IAM-Administrator kann die Berechtigungen für Service-verknüpfte Rollen anzeigen, aber nicht bearbeiten.

Details zum Erstellen oder Verwalten von serviceverknüpften Rollen in Amazon Managed Service für Prometheus finden Sie unter Verwenden von serviceverknüpften Rollen für Amazon Managed Service für Prometheus.

Beispiele für identitätsbasierte Richtlinien für Amazon Managed Service für Prometheus

Benutzer und Rollen besitzen standardmäßig keine Berechtigungen zum Erstellen oder Ändern von Ressourcen in Amazon Managed Service für Prometheus. Ein IAM-Administrator muss IAM-Richtlinien erstellen, die Benutzern die Berechtigung erteilen, Aktionen für die Ressourcen auszuführen, die sie benötigen.

Informationen dazu, wie Sie unter Verwendung dieser beispielhaften JSON-Richtliniendokumente eine identitätsbasierte IAM-Richtlinie erstellen, finden Sie unter Erstellen von IAM-Richtlinien (Konsole) im IAM-Benutzerhandbuch.

Einzelheiten zu den Aktionen und Ressourcentypen, die von Amazon Managed Service für Prometheus definiert wurden, einschließlich des Formats ARNs für die einzelnen Ressourcentypen, finden Sie unter Aktionen, Ressourcen und Bedingungsschlüssel für Amazon Managed Service for Prometheus in der Service Authorization Reference.

Themen

- Best Practices für Richtlinien
- Verwenden der Konsole von Amazon Managed Service für Prometheus
- Gewähren der Berechtigung zur Anzeige der eigenen Berechtigungen für Benutzer

Best Practices für Richtlinien

Identitätsbasierte Richtlinien können festlegen, ob jemand Ressourcen in Amazon Managed Service für Prometheus in Ihrem Konto erstellen, darauf zugreifen oder löschen kann. Dies kann zusätzliche Kosten für Ihr verursachen AWS-Konto. Wenn Sie identitätsbasierte Richtlinien erstellen oder bearbeiten, befolgen Sie diese Richtlinien und Empfehlungen:

- Erste Schritte mit AWS verwalteten Richtlinien und Umstellung auf Berechtigungen mit den geringsten Rechten — Verwenden Sie die AWS verwalteten Richtlinien, die Berechtigungen für viele gängige Anwendungsfälle gewähren, um damit zu beginnen, Ihren Benutzern und Workloads Berechtigungen zu gewähren. Sie sind in Ihrem verfügbar. AWS-Konto Wir empfehlen Ihnen, die Berechtigungen weiter zu reduzieren, indem Sie vom AWS Kunden verwaltete Richtlinien definieren, die speziell auf Ihre Anwendungsfälle zugeschnitten sind. Weitere Informationen finden Sie unter Von AWS verwaltete Richtlinien oder Von AWS verwaltete Richtlinien für Auftragsfunktionen im IAM-Benutzerhandbuch.

- Anwendung von Berechtigungen mit den geringsten Rechten – Wenn Sie mit IAM-Richtlinien Berechtigungen festlegen, gewähren Sie nur die Berechtigungen, die für die Durchführung einer Aufgabe erforderlich sind. Sie tun dies, indem Sie die Aktionen definieren, die für bestimmte Ressourcen unter bestimmten Bedingungen durchgeführt werden können, auch bekannt als die geringsten Berechtigungen. Weitere Informationen zur Verwendung von IAM zum Anwenden von Berechtigungen finden Sie unter Richtlinien und Berechtigungen in IAM im IAM-Benutzerhandbuch.
- Verwenden von Bedingungen in IAM-Richtlinien zur weiteren Einschränkung des Zugriffs – Sie können Ihren Richtlinien eine Bedingung hinzufügen, um den Zugriff auf Aktionen und Ressourcen zu beschränken. Sie können beispielsweise eine Richtlinienbedingung schreiben, um festzulegen, dass alle Anforderungen mithilfe von SSL gesendet werden müssen. Sie können auch Bedingungen verwenden, um Zugriff auf Serviceaktionen zu gewähren, wenn diese für einen bestimmten Zweck verwendet werden AWS-Service, z. CloudFormation B. Weitere Informationen finden Sie unter IAM-JSON-Richtlinienelemente: Bedingung im IAM-Benutzerhandbuch.
- Verwenden von IAM Access Analyzer zur Validierung Ihrer IAM-Richtlinien, um sichere und funktionale Berechtigungen zu gewährleisten – IAM Access Analyzer validiert neue und vorhandene Richtlinien, damit die Richtlinien der IAM-Richtliniensprache (JSON) und den bewährten IAM-Methoden entsprechen. IAM Access Analyzer stellt mehr als 100 Richtlinienprüfungen und umsetzbare Empfehlungen zur Verfügung, damit Sie sichere und funktionale Richtlinien erstellen können. Weitere Informationen finden Sie unter Richtlinienvalidierung mit IAM Access Analyzer im IAM-Benutzerhandbuch.
- Multi-Faktor-Authentifizierung (MFA) erforderlich — Wenn Sie ein Szenario haben, das IAM-Benutzer oder einen Root-Benutzer in Ihrem System erfordert AWS-Konto, aktivieren Sie MFA für zusätzliche Sicherheit. Um MFA beim Aufrufen von API-Vorgängen anzufordern, fügen Sie Ihren Richtlinien MFA-Bedingungen hinzu. Weitere Informationen finden Sie unter Sicherer API-Zugriff mit MFA im IAM-Benutzerhandbuch.

Weitere Informationen zu bewährten Methoden in IAM finden Sie unter Best Practices für die Sicherheit in IAM im IAM-Benutzerhandbuch.

Verwenden der Konsole von Amazon Managed Service für Prometheus

Um auf die Konsole von Amazon Managed Service für Prometheus zugreifen zu können, müssen Sie über einen Mindestsatz von Berechtigungen verfügen. Diese Berechtigungen müssen es Ihnen ermöglichen, Details über die Ressourcen in Amazon Managed Service für Prometheus in Ihrem AWS-Konto aufzulisten und anzuzeigen. Wenn Sie eine identitätsbasierte Richtlinie erstellen, die

strenger ist als die mindestens erforderlichen Berechtigungen, funktioniert die Konsole nicht wie vorgesehen für Entitäten (Benutzer oder Rollen) mit dieser Richtlinie.

Sie müssen Benutzern, die nur die API oder die API aufrufen, keine Mindestberechtigungen für die AWS CLI Konsole gewähren. AWS Stattdessen sollten Sie nur Zugriff auf die Aktionen zulassen, die der API-Operation entsprechen, die die Benutzer ausführen möchten.

Um sicherzustellen, dass Benutzer und Rollen weiterhin die Amazon Managed Service for Prometheus-Konsole verwenden können, fügen Sie den Entitäten auch den Amazon Managed Service für Prometheus `ConsoleAccess` oder die `ReadOnly` AWS verwaltete Richtlinie hinzu. Weitere Informationen finden Sie unter Hinzufügen von Berechtigungen zu einem Benutzer im IAM-Benutzerhandbuch.

Gewähren der Berechtigung zur Anzeige der eigenen Berechtigungen für Benutzer

In diesem Beispiel wird gezeigt, wie Sie eine Richtlinie erstellen, die IAM-Benutzern die Berechtigung zum Anzeigen der eingebundenen Richtlinien und verwalteten Richtlinien gewährt, die ihrer Benutzeridentität angefügt sind. Diese Richtlinie beinhaltet Berechtigungen zum Ausführen dieser Aktion auf der Konsole oder programmgesteuert mithilfe der OR-API. AWS CLI 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": "*"
        }
    ]
}
```

Problembehebung von Identität und Zugriff auf Amazon Managed Service für Prometheus

Diagnostizieren und beheben Sie mithilfe der folgenden Informationen gängige Probleme, die bei der Verwendung von Amazon Managed Service für Prometheus und IAM auftreten können.

Themen

- Ich bin nicht autorisiert, eine Aktion in Amazon Managed Service für Prometheus auszuführen.
- Ich bin nicht berechtigt, iam auszuführen: PassRole
- Ich möchte Personen außerhalb meines AWS Kontos den Zugriff auf meine Amazon Managed Service for Prometheus-Ressourcen ermöglichen

Ich bin nicht autorisiert, eine Aktion in Amazon Managed Service für Prometheus auszuführen.

Wenn Sie eine Fehlermeldung erhalten, dass Sie nicht zur Durchführung einer Aktion berechtigt sind, müssen Ihre Richtlinien aktualisiert werden, damit Sie die Aktion durchführen können.

Der folgende Beispielfehler tritt auf, wenn der IAM-Benutzer `mateojackson` versucht, über die Konsole Details zu einer fiktiven *`my-example-widget`*-Ressource anzuzeigen, jedoch nicht über `aps:`*`GetWidget`*-Berechtigungen verfügt.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 aps:GetWidget on resource: my-example-widget
```

In diesem Fall muss die Richtlinie für den Benutzer `mateojackson` aktualisiert werden, damit er mit der `aps:GetWidget`-Aktion auf die *`my-example-widget`*-Ressource zugreifen kann.

Wenn Sie Hilfe benötigen, wenden Sie sich an Ihren AWS Administrator. Ihr Administrator hat Ihnen Ihre Anmeldeinformationen zur Verfügung gestellt.

Ich bin nicht berechtigt, iam auszuführen: PassRole

Wenn Sie die Fehlermeldung erhalten, dass Sie nicht zur Ausführung der `iam:PassRole`-Aktion autorisiert sind, müssen Ihre Richtlinien aktualisiert werden, um eine Rolle an Amazon Managed Service für Prometheus übergeben zu können.

Einige AWS-Services ermöglichen es Ihnen, eine bestehende Rolle an diesen Dienst zu übergeben, anstatt eine neue Servicerolle oder eine dienstverknüpfte Rolle zu erstellen. Hierzu benötigen Sie Berechtigungen für die Übergabe der Rolle an den Dienst.

Der folgende Beispielfehler tritt auf, wenn ein IAM-Benutzer mit dem Namen `marymajor` versucht, die Konsole zu verwenden, um eine Aktion in Amazon Managed Service für Prometheus auszuführen. Die Aktion erfordert jedoch, dass der Service über Berechtigungen verfügt, die durch eine Servicerolle gewährt werden. Mary besitzt keine Berechtigungen für die Übergabe der Rolle an den Dienst.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

In diesem Fall müssen die Richtlinien von Mary aktualisiert werden, um die Aktion `iam:PassRole` ausführen zu können.

Wenn Sie Hilfe benötigen, wenden Sie sich an Ihren AWS Administrator. Ihr Administrator hat Ihnen Ihre Anmeldeinformationen zur Verfügung gestellt.

Ich möchte Personen außerhalb meines AWS Kontos den Zugriff auf meine Amazon Managed Service for Prometheus-Ressourcen ermöglichen

Sie können eine Rolle erstellen, mit der Benutzer in anderen Konten oder Personen außerhalb Ihrer Organisation auf Ihre Ressourcen zugreifen können. Sie können festlegen, wem die Übernahme der Rolle anvertraut wird. Für Dienste, die ressourcenbasierte Richtlinien oder Zugriffskontrolllisten (ACLs) unterstützen, können Sie diese Richtlinien verwenden, um Personen Zugriff auf Ihre Ressourcen zu gewähren.

Weitere Informationen dazu finden Sie hier:

- Informationen dazu, ob Amazon Managed Service für Prometheus diese Features unterstützt, finden Sie unter So funktioniert Amazon Managed Service für Prometheus mit IAM.
- Informationen dazu, wie Sie Zugriff auf Ihre Ressourcen in AWS-Konten Ihrem Besitz gewähren können, finden Sie im IAM-Benutzerhandbuch unter Gewähren des Zugriffs für einen IAM-Benutzer in einem anderen AWS-Konto , dem Sie gehören.
- Informationen dazu, wie Sie Dritten Zugriff auf Ihre Ressourcen gewähren können AWS-Konten, finden Sie AWS-Konten im IAM-Benutzerhandbuch unter Gewähren des Zugriffs für Dritte.
- Informationen dazu, wie Sie über einen Identitätsverbund Zugriff gewähren, finden Sie unter Gewähren von Zugriff für extern authentifizierte Benutzer (Identitätsverbund) im IAM-Benutzerhandbuch.
- Informationen zum Unterschied zwischen der Verwendung von Rollen und ressourcenbasierten Richtlinien für den kontoübergreifenden Zugriff finden Sie unter Kontoübergreifender Ressourcenzugriff in IAM im IAM-Benutzerhandbuch.

IAM-Berechtigungen und Richtlinien

Für den Zugriff auf Aktionen und Daten von Amazon Managed Service für Prometheus sind Anmeldeinformationen erforderlich. Diese Anmeldeinformationen müssen über Berechtigungen verfügen, um die Aktionen auszuführen und auf die AWS Ressourcen zuzugreifen, z. B. zum Abrufen von Amazon Managed Service for Prometheus-Daten über Ihre Cloud-Ressourcen. In den folgenden Abschnitten erfahren Sie, wie Sie AWS Identity and Access Management (IAM) und Amazon Managed Service for Prometheus verwenden können, um Ihre Ressourcen zu schützen, indem Sie kontrollieren, wer darauf zugreifen kann. Weitere Informationen finden Sie unter Richtlinien und Berechtigungen in IAM.

Berechtigungen für Amazon Managed Service für Prometheus

Eine Liste der möglichen Amazon Managed Service for Prometheus Aktionen, Ressourcentypen und Bedingungsschlüssel finden Sie unter Aktionen, Ressourcen und Bedingungsschlüssel für Amazon Managed Service for Prometheus.

Beispiel für IAM-Richtlinien

Dieser Abschnitt enthält Beispiele für andere selbstverwaltete Richtlinien, die Sie erstellen können.

Die folgende IAM-Richtlinie gewährt vollen Zugriff auf Amazon Managed Service für Prometheus und ermöglicht es einem Benutzer außerdem, Amazon-EKS-Cluster zu entdecken und deren Details einzusehen.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "aps:*",
                "eks:DescribeCluster",
                "eks:ListClusters"
            ],
            "Resource": "*"
        }
    ]
}
```

Compliance-Validierung für Amazon Managed Service für Prometheus

Unser neues AWS Anmeldeerlebnis ist nicht für regulierte Workloads konzipiert. Wenn Sie unser neues AWS Anmeldeerlebnis nutzen, es aber AWS für regulierte Workloads nutzen möchten, können Sie sich für erweiterte Funktionen für AWS Ihre Umgebung anmelden oder diese aktivieren. AWS

Um zu erfahren, ob AWS-Service ein Programm AWS-Services in den Geltungsbereich bestimmter Compliance-Programme fällt, finden Sie unter Umfang nach Compliance-Programm und wählen Sie das Compliance-Programm aus, an dem Sie interessiert sind. Allgemeine Informationen finden Sie unter AWS Compliance-Programme, .

Sie können Auditberichte von Drittanbietern mit herunterladen AWS Artifact. Weitere Informationen finden Sie unter Berichte herunterladen unter Berichte AWS Artifact .

Ihre Verantwortung für die Einhaltung der Vorschriften bei der Nutzung AWS-Services hängt von der Vertraulichkeit Ihrer Daten, den Compliance-Zielen Ihres Unternehmens sowie den geltenden

Gesetzen und Vorschriften ab. Weitere Informationen zu Ihrer Compliance-Verantwortung bei der Verwendung AWS-Services finden Sie in der AWS Sicherheitsdokumentation.

Ausfallsicherheit in Amazon Managed Service für Prometheus

Die AWS globale Infrastruktur basiert auf AWS Regionen und Availability Zones. AWS Regionen bieten mehrere physisch getrennte und isolierte Availability Zones, die über Netzwerke mit niedriger Latenz, hohem Durchsatz und hoher Redundanz miteinander verbunden sind. Mithilfe von Availability Zones können Sie Anwendungen und Datenbanken erstellen und ausführen, die automatisch Failover zwischen Zonen ausführen, ohne dass es zu Unterbrechungen kommt. Availability Zones sind besser verfügbar, fehlertoleranter und skalierbarer als herkömmliche Infrastrukturen mit einem oder mehreren Rechenzentren.

Weitere Informationen zu AWS Regionen und Availability Zones finden Sie unter AWS Globale Infrastruktur.

Zusätzlich zur AWS globalen Infrastruktur bietet Amazon Managed Service for Prometheus mehrere Funktionen, um Ihre Anforderungen an Datenstabilität und Datensicherung zu erfüllen, einschließlich Unterstützung für Daten mit hoher Verfügbarkeit.

Sicherheit der Infrastruktur in Amazon Managed Service für Prometheus

Als verwalteter Service ist Amazon Managed Service for Prometheus durch AWS globale Netzwerksicherheit geschützt. Informationen zu AWS Sicherheitsdiensten und zum AWS Schutz der Infrastruktur finden Sie unter AWS Cloud-Sicherheit. Informationen zum Entwerfen Ihrer AWS Umgebung unter Verwendung der bewährten Methoden für die Infrastruktursicherheit finden Sie unter Infrastructure Protection in Security Pillar AWS Well-Architected Framework.

Sie verwenden AWS veröffentlichte API-Aufrufe, um über das Netzwerk auf Amazon Managed Service for Prometheus zuzugreifen. Kunden müssen Folgendes unterstützen:

- Transport Layer Security (TLS). Wir benötigen TLS 1.2 und empfehlen TLS 1.3.
- Verschlüsselungs-Suiten mit Perfect Forward Secrecy (PFS) wie DHE (Ephemeral Diffie-Hellman) oder ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). Die meisten modernen Systeme wie Java 7 und höher unterstützen diese Modi.

Verwenden von serviceverknüpften Rollen für Amazon Managed Service für Prometheus

Amazon Managed Service für Prometheus-Anwendungen AWS Identity and Access Management (ICH BIN) dienstgebundene Rollen. Eine serviceverknüpfte Rolle ist eine einzigartige Art von IAM-Rolle, die direkt mit Amazon Managed Service for Prometheus verknüpft ist. Service-linked Rollen werden von Amazon Managed Service for Prometheus vordefiniert und beinhalten alle Berechtigungen, die der Service benötigt, um andere Rollen aufzurufen AWS Dienstleistungen in Ihrem Namen.

Eine serviceverknüpfte Rolle macht die Einrichtung von Amazon Managed Service für Prometheus einfacher, da Sie die erforderlichen Berechtigungen nicht manuell hinzufügen müssen. Amazon Managed Service für Prometheus definiert die Berechtigungen seiner mit dem Service verbundenen Rollen, und sofern nicht anders definiert, kann nur Amazon Managed Service für Prometheus seine Rollen übernehmen. Die definierten Berechtigungen umfassen die Vertrauens- und Berechtigungsrichtlinie. Diese Berechtigungsrichtlinie kann keinen anderen IAM-Entitäten zugewiesen werden.

Verwenden von Rollen zum Abrufen von Metriken aus EKS

Beim automatischen Scraping von Metriken mithilfe von Amazon Managed Service for Prometheus Managed Collector wird der AWSServiceRoleForAmazonPrometheusScraper Die service-verknüpfte Rolle wird verwendet, um die Einrichtung von Managed Collector zu vereinfachen, da Sie die erforderlichen Berechtigungen nicht manuell hinzufügen müssen. Amazon Managed Service für Prometheus definiert die Berechtigungen und nur Amazon Managed Service für Prometheus kann die Rolle übernehmen.

Informationen zu anderen Diensten, die serviceverknüpfte Rollen unterstützen, finden Sie unter AWS Dienste, die mit IAM funktionieren und suchen Sie nach den Diensten, die Ja in der Service-linked Rollen Spalte. Wählen Sie über einen Link Ja aus, um die Dokumentation zu einer serviceverknüpften Rolle für diesen Service anzuzeigen.

Service-linked Rollenberechtigungen für Amazon Managed Service für Prometheus

Amazon Managed Service für Prometheus verwendet eine serviceverknüpfte Rolle, die mit dem Präfix benannt ist AWSServiceRoleForAmazonPrometheusScraper damit Amazon Managed Service for Prometheus automatisch Metriken in Ihren Amazon EKS-Clustern scrapen kann.

Das AWSServiceRoleForAmazonPrometheusScraper Die dienstgebundene Rolle vertraut darauf, dass die folgenden Dienste die Rolle übernehmen:

- `scraper.aps.amazonaws.com`

Die Richtlinie für Rollenberechtigungen heißt AmazonPrometheusScraperServiceRolePolicy ermöglicht Amazon Managed Service for Prometheus, die folgenden Aktionen für die angegebenen Ressourcen durchzuführen:

- Lesen und ändern Sie die Netzwerkkonfiguration, um eine Verbindung zu dem Netzwerk herzustellen, das Ihren Amazon EKS-Cluster enthält.
- Lesen Sie Metriken aus Amazon EKS-Clustern und schreiben Sie Metriken in Ihre Amazon Managed Service for Prometheus Workspaces.
- Schreiben Sie Metriken in Amazon CloudWatch Metrik-Datensätze im selben Konto.

Die Rolle hat standardmäßig keine Amazon MSK-Berechtigungen. Um Kennzahlen für Verbrauchergruppen mit einem verwalteten Kafka-Exporter zu erfassen, gewähren Sie der Rolle Zugriff mithilfe einer Amazon MSK-Cluster-Richtlinie. Weitere Informationen finden Sie unter Gewähren Sie dem Scraper Zugriff auf den Cluster.

Sie müssen Berechtigungen konfigurieren, damit Ihre Benutzer, Gruppen oder Rollen eine serviceverknüpfte Rolle erstellen können. Weitere Informationen finden Sie unter Service-linked Rollenberechtigungen in der IAM-Benutzerhandbuch.

Erstellen einer serviceverknüpften Rolle für Amazon Managed Service für Prometheus

Sie müssen eine serviceverknüpfte Rolle nicht manuell erstellen. Wenn Sie eine verwaltete Collector-Instance mithilfe von Amazon EKS oder Amazon Managed Service for Prometheus in der AWS-Managementkonsole, der AWS CLI, oder der AWS API, Amazon Managed Service for Prometheus erstellt die serviceverknüpfte Rolle für Sie.

> **Important**
>
> Diese serviceverknüpfte Rolle kann in Ihrem Konto erscheinen, wenn Sie eine Aktion in einem anderen Service abgeschlossen haben, der die von dieser Rolle unterstützten Features verwendet. Weitere Informationen finden Sie unter Eine neue Rolle erschien in meinem AWS-Konto.

Wenn Sie diese serviceverknüpfte Rolle löschen und sie dann erneut erstellen müssen, können Sie dasselbe Verfahren anwenden, um die Rolle in Ihrem Konto neu anzulegen. Wenn Sie eine verwaltete Kollektor-Instance erstellen, indem Sie Amazon EKS oder Amazon Managed Service für Prometheus erstellen, erstellt Amazon Managed Service für Prometheus die serviceverknüpfte Rolle für Sie.

Bearbeiten einer serviceverknüpften Rolle für Amazon Managed Service für Prometheus

Amazon Managed Service für Prometheus erlaubt Ihnen nicht, das zu bearbeiten AWSServiceRoleForAmazonPrometheusScraper Rolle, die mit einem Service verknüpft ist. Da möglicherweise verschiedene Entitäten auf die Rolle verweisen, kann der Rollenname nach dem Erstellen einer serviceverknüpften Rolle nicht mehr geändert werden. Sie können jedoch die Beschreibung der Rolle mit IAM bearbeiten. Weitere Informationen finden Sie unter Bearbeiten einer serviceverknüpften Rolle im IAM-Benutzerhandbuch.

Löschen einer serviceverknüpften Rolle für Amazon Managed Service für Prometheus

Sie müssen das nicht manuell löschen AWSServiceRoleForAmazonPrometheusScraper Rolle. Wenn Sie alle verwalteten Collector-Instanzen löschen, die der Rolle in der AWS-Managementkonsole, der AWS CLI, oder der AWS API, Amazon Managed Service for Prometheus bereinigt die Ressourcen und löscht die serviceverknüpfte Rolle für Sie.

Unterstützte Regionen für serviceverknüpfte Rollen für Amazon Managed Service für Prometheus

Amazon Managed Service für Prometheus unterstützt die Verwendung von serviceverknüpften Rollen in allen Regionen, in denen der Service verfügbar ist. Weitere Informationen finden Sie unter Unterstützte Regionen.

Protokollieren von Amazon Managed Service for Prometheus API-Aufrufen mithilfe von AWS CloudTrail

Amazon Managed Service for Prometheus ist in einen Service integriert AWS CloudTrail, der eine Aufzeichnung der von einem Benutzer, einer Rolle oder einem ausgeführten Aktionen bereitstellt. AWS-Service CloudTrailerfasst alle API-Aufrufe für Amazon Managed Service for Prometheus als Ereignisse. Zu den erfassten Aufrufen gehören Aufrufe von der Amazon Managed Service for

Prometheus-Konsole und Code-Aufrufe an die API-Operationen von Amazon Managed Service for Prometheus. Anhand der von gesammelten Informationen können Sie die Anfrage CloudTrail, die an Amazon Managed Service for Prometheus gestellt wurde, die IP-Adresse, von der aus die Anfrage gestellt wurde, wann sie gestellt wurde, und weitere Details ermitteln.

Jeder Ereignis- oder Protokolleintrag enthält Informationen zu dem Benutzer, der die Anforderung generiert hat. Die Identitätsinformationen unterstützen Sie bei der Ermittlung der folgenden Punkte:

- Ob die Anforderung mit Anmeldeinformationen des Root-Benutzers oder des Benutzers gestellt wurde.
- Die Anforderung wurde im Namen eines IAM-Identity-Center-Benutzers erstellt.
- Gibt an, ob die Anforderung mit temporären Sicherheitsanmeldeinformationen für eine Rolle oder einen Verbundbenutzer gesendet wurde.
- Ob die Anforderung aus einem anderen AWS-Service gesendet wurde.

CloudTrail ist in Ihrem aktiv AWS-Konto , wenn Sie das Konto erstellen, und Sie haben automatisch Zugriff auf den CloudTrail Event-Verlauf. Der CloudTrail Ereignisverlauf bietet eine einsehbare, durchsuchbare, herunterladbare und unveränderliche Aufzeichnung der aufgezeichneten Verwaltungsereignisse der letzten 90 Tage in einem. AWS-Region Weitere Informationen finden Sie im AWS CloudTrail Benutzerhandbuch unter Arbeiten mit dem CloudTrail Ereignisverlauf. Für die Anzeige des Eventverlaufs CloudTrail fallen keine Gebühren an.

Für eine fortlaufende Aufzeichnung der Ereignisse in AWS-Konto den letzten 90 Tagen erstellen Sie einen Trail- oder CloudTrail Lake-Event-Datenspeicher.

CloudTrail Pfade

Ein Trail ermöglicht CloudTrail die Übermittlung von Protokolldateien an einen Amazon S3 S3-Bucket. Alle mit dem erstellten Pfade AWS-Managementkonsole sind regionsübergreifend. Sie können mithilfe von AWS CLI einen Einzel-Region- oder einen Multi-Region-Trail erstellen. Es wird empfohlen, einen Trail mit mehreren Regionen zu erstellen, da Sie alle Aktivitäten AWS-Regionen in Ihrem Konto erfassen. Wenn Sie einen Einzel-Region-Trail erstellen, können Sie nur die Ereignisse anzeigen, die im AWS-Region des Trails protokolliert wurden. Weitere Informationen zu Trails finden Sie unter Erstellen eines Trails für Ihr AWS-Konto und Erstellen eines Trails für eine Organisation im AWS CloudTrail -Benutzerhandbuch.

Sie können eine Kopie Ihrer laufenden Verwaltungsereignisse kostenlos an Ihren Amazon S3 S3-Bucket senden, CloudTrail indem Sie einen Trail erstellen. Es fallen jedoch Amazon S3 S3-

Speichergebühren an. Weitere Informationen zur CloudTrail Preisgestaltung finden Sie unter AWS CloudTrail Preise. Informationen zu Amazon-S3-Preisen finden Sie unter Amazon S3 – Preise.

CloudTrail Datenspeicher für Ereignisse in Lake

CloudTrail Mit Lake können Sie SQL-basierte Abfragen für Ihre Ereignisse ausführen. CloudTrail Lake konvertiert bestehende Ereignisse im zeilenbasierten JSON-Format in das Apache ORC-Format. ORC ist ein spaltenförmiges Speicherformat, das für den schnellen Abruf von Daten optimiert ist. Die Ereignisse werden in Ereignisdatenspeichern zusammengefasst, bei denen es sich um unveränderliche Sammlungen von Ereignissen handelt, die auf Kriterien basieren, die Sie mit Hilfe von erweiterten Ereignisselektoren auswählen. Die Selektoren, die Sie auf einen Ereignisdatenspeicher anwenden, steuern, welche Ereignisse bestehen bleiben und für Sie zur Abfrage verfügbar sind. Weitere Informationen zu CloudTrail Lake finden Sie unter Arbeiten mit AWS CloudTrail Lake im AWS CloudTrail Benutzerhandbuch.

CloudTrail Für das Speichern und Abfragen von Ereignisdaten in Lake fallen Kosten an. Beim Erstellen eines Ereignisdatenspeichers wählen Sie die Preisoption aus, die für den Ereignisdatenspeicher genutzt werden soll. Die Preisoption bestimmt die Kosten für die Erfassung und Speicherung von Ereignissen sowie die standardmäßige und maximale Aufbewahrungsdauer für den Ereignisdatenspeicher. Weitere Informationen zur CloudTrail Preisgestaltung finden Sie unter AWS CloudTrail Preise.

Verwaltungsveranstaltungen von Amazon Managed Service für Prometheus in CloudTrail

Verwaltungsereignisse bieten Informationen über Verwaltungsvorgänge, die mit Ressourcen in Ihrem System ausgeführt werden. AWS-Konto Sie werden auch als Vorgänge auf Steuerebene bezeichnet. CloudTrail Protokolliert standardmäßig Verwaltungsereignisse.

Amazon Managed Service for Prometheus protokolliert alle Vorgänge auf der Amazon Managed Service for Prometheus Kontrollebene als Verwaltungsereignisse. Eine Liste der Vorgänge auf der Amazon Managed Service for Prometheus-Steuerebene, die Amazon Managed Service for Prometheus protokolliert CloudTrail, finden Sie in der Amazon Managed Service for Prometheus API-Referenz.

Beispiele für Veranstaltungen mit Amazon Managed Service für Prometheus

Ein Ereignis stellt eine einzelne Anfrage aus einer beliebigen Quelle dar und enthält Informationen über den angeforderten API-Vorgang, Datum und Uhrzeit des Vorgangs, Anforderungsparameter usw. CloudTrail Protokolldateien sind kein geordneter Stack-Trace der öffentlichen API-Aufrufe, sodass Ereignisse nicht in einer bestimmten Reihenfolge angezeigt werden.

Beispiel: CreateWorkspace

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag, der die CreateWorkspace Aktion demonstriert.


```
{
    "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"
}
```

Beispiel: CreateAlertManagerDefinition

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag, der die CreateAlertManagerDefinition Aktion demonstriert.

```
{
    "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"
}
```


Beispiel: CreateRuleGroupsNamespace

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag, der die CreateRuleGroupsNamespace Aktion demonstriert.


```
{
    "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"
}
```


Informationen zu CloudTrail Datensatzinhalten finden Sie im AWS CloudTrail Benutzerhandbuch unter CloudTrail Datensatzinhalte.

IAM-Rollen für Servicekonten einrichten

Mit IAM-Rollen für Servicekonten können Sie eine IAM-Rolle mit einem Kubernetes-Servicekonto verknüpfen. Dieses Dienstkonto kann dann AWS Berechtigungen für die Container in jedem Pod gewähren, der dieses Dienstkonto verwendet. Weitere Informationen finden Sie unter IAM-Rollen für Servicekonten.

IAM-Rollen für Servicekonten werden auch als Servicerollen bezeichnet.

In Amazon Managed Service für Prometheus können Sie mithilfe von Servicerollen die Rollen erhalten, die Sie für die Autorisierung und Authentifizierung zwischen Amazon Managed Service für Prometheus, Prometheus-Servern und Grafana-Servern benötigen.

Voraussetzungen

Für die Verfahren auf dieser Seite müssen Sie die Befehlszeilenschnittstelle AWS CLI und die EKSCTL-Befehlszeilenschnittstelle installiert haben.

Einrichten von Servicerollen für die Erfassung von Metriken aus Amazon-EKS-Clustern

Um die Servicerollen so einzurichten, dass Amazon Managed Service für Prometheus Metriken von Prometheus-Servern in Amazon-EKS-Clustern erfassen kann, müssen Sie bei einem Konto mit den folgenden Berechtigungen angemeldet sein:

- `iam:CreateRole`
- `iam:CreatePolicy`
- `iam:GetRole`
- `iam:AttachRolePolicy`
- `iam:GetOpenIDConnectProvider`

So richten Sie die Servicerolle für die Erfassung in Amazon Managed Service für Prometheus ein

1. Erstellen Sie eine Datei mit dem Namen `createIRSA-AMPIngest.sh` und dem folgenden Inhalt. Ersetzen Sie <my_amazon_eks_clustername> durch den Namen Ihres Clusters und ersetzen Sie <my_prometheus_namespace> durch Ihren Prometheus-Namespace.

```
#!/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. Geben Sie den folgenden Befehl ein, um dem Skript die erforderlichen Rechte zu erteilen.

```
chmod +x createIRSA-AMPIngest.sh
```

3. Führen Sie das Skript aus.

Richten Sie IAM-Rollen für Servicekonten zur Abfrage von Metriken ein

Zum Einrichten der IAM-Rolle für das Servicekonto (Service-Rolle), um die Abfrage von Metriken aus den Workspaces von Amazon Managed Service für Prometheus zu ermöglichen, müssen Sie an einem Konto mit den folgenden Berechtigungen angemeldet sein:

- `iam:CreateRole`
- `iam:CreatePolicy`
- `iam:GetRole`
- `iam:AttachRolePolicy`
- `iam:GetOpenIDConnectProvider`

Um Servicerollen für die Abfrage von Metriken von Amazon Managed Service für Prometheus einzurichten;

1. Erstellen Sie eine Datei mit dem Namen `createIRSA-AMPQuery.sh` und dem folgenden Inhalt. Ersetzen Sie <my_amazon_eks_clustername> durch den Namen Ihres Clusters und <my_prometheus_namespace> durch Ihren Prometheus-Namespace.

```
#!/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. Geben Sie den folgenden Befehl ein, um dem Skript die erforderlichen Rechte zu erteilen.

```
chmod +x createIRSA-AMPQuery.sh
```

3. Führen Sie das Skript aus.

Verwendung von Amazon Managed Service für Prometheus mit Schnittstellen-VPC-Endpunkten

Wenn Sie Amazon Virtual Private Cloud (Amazon VPC) zum Hosten Ihrer AWS Ressourcen verwenden, können Sie private Verbindungen zwischen Ihrer VPC und Amazon Managed Service for Prometheus herstellen. Sie können diese Verbindungen verwenden, damit Amazon Managed Service für Prometheus mit den Ressourcen in der VPC kommunizieren kann, ohne das öffentliche Internet verwenden zu müssen.

Amazon VPC ist ein AWS Service, mit dem Sie AWS Ressourcen in einem von Ihnen definierten virtuellen Netzwerk starten können. Mit einer VPC haben Sie die Kontrolle über Ihre Netzwerkeinstellungen, wie IP-Adressbereich, Subnetze, Routing-Tabellen und Netzwerk-Gateways. Um Ihre VPC mit Amazon Managed Service für Prometheus zu verbinden, definieren Sie einen Schnittstellen-VPC-Endpunkt, um Ihre VPC mit AWS Services zu verbinden. Der Endpunkt bietet zuverlässige, skalierbare Konnektivität zu Amazon Managed Service für Prometheus, ohne dass ein Internet-Gateway, eine NAT-Instance (Network Address Translation) oder eine VPN-Verbindung erforderlich ist. Weitere Informationen finden Sie unter Was ist Amazon VPC im Benutzerhandbuch zu Amazon VPC.

Schnittstellen-VPC-Endpunkte werden von einer AWS Technologie unterstützt AWS PrivateLink, die die private Kommunikation zwischen AWS Diensten über eine elastische Netzwerkschnittstelle mit privaten IP-Adressen ermöglicht. Weitere Informationen finden Sie im Blogbeitrag Neu — AWS PrivateLink für AWS Dienste.

Die folgenden Informationen sind für Benutzer von Amazon VPC vorgesehen. Informationen zu ersten Schritten mit Amazon VPC finden Sie unter Erste Schritte im Amazon-VPC-Benutzerhandbuch.

Erstellen eines Schnittstellen-VPC-Endpunkts für Amazon Managed Service für Prometheus

Erstellen eines Schnittstellen-VPC-Endpunkts, um mit der Nutzung von Amazon Managed Service für Prometheus zu beginnen. Wählen Sie aus den folgenden Endpunkten mit Servicenamen:

- `com.amazonaws.region.aps-workspaces`

  Wählen Sie diesen Dienstnamen, um mit Prometheus-compatible APIs zu arbeiten. Weitere Informationen finden Sie unter Prometheus-compatible APIs im Amazon Managed Service für Prometheus-Benutzerhandbuch.

- `com.amazonaws.region.aps`

  Wählen Sie diesen Servicenamen, um Workspace-Management-Aufgaben auszuführen. Weitere Informationen finden Sie unter APIs für Amazon Managed Service für Prometheus im Benutzerhandbuch von Amazon Managed Service für Prometheus.

> **Note**
>
> Wenn Sie remote_write in einer VPC ohne direkten Internetzugang verwenden, müssen Sie auch einen Schnittstellen-VPC-Endpunkt für erstellen AWS -Security-Token-Service, damit sigv4 den Endpunkt verarbeiten kann. Informationen zum Erstellen eines VPC-Endpunkts für AWS STS finden Sie unter Verwenden von AWS STS Schnittstellen-VPC-Endpunkten im Benutzerhandbuch. AWS Identity and Access Management Sie müssen festlegen, dass regionalisierte AWS STS Endpunkte verwendet werden.

Weitere Informationen, einschließlich schrittweiser Anleitungen zum Erstellen eines Schnittstellen-VPC-Endpunkts, finden Sie unter Erstellen eines Schnittstellen-Endpunkts im Amazon-VPC-Benutzerhandbuch.

> **Note**
>
> Sie können VPC-Endpunktrichtlinien verwenden, um den Zugriff auf Ihren VPC-Endpunkt der Schnittstelle von Amazon Managed Service für Prometheus zu kontrollieren. Weitere Informationen finden Sie im nächsten Abschnitt.

Wenn Sie einen Schnittstellen-VPC-Endpunkt für Amazon Managed Service für Prometheus erstellt haben und bereits an die Workspaces in Ihrer VPC Daten senden, werden die Metriken standardmäßig über den Schnittstellen-VPC-Endpunkt gesendet. Amazon Managed Service für Prometheus verwendet öffentliche Endpunkte oder private Schnittstellenendpunkte (je nachdem, welche verwendet werden), um diese Aufgabe auszuführen.

Steuern des Zugriffs auf Ihren VPC-Endpunkt von Amazon Managed Service für Prometheus

Sie können VPC-Endpunktrichtlinien verwenden, um den Zugriff auf Ihren VPC-Endpunkt der Schnittstelle von Amazon Managed Service für Prometheus zu kontrollieren. Eine VPC-Endpunktrichtlinie ist eine IAM-Ressourcenrichtlinie, die Sie einem Endpunkt beim Erstellen oder Ändern des Endpunkts zuordnen. Wenn Sie einem Endpunkt beim Erstellen keine Richtlinie zuordnen, ordnet Amazon VPC ihm eine Standardrichtlinie mit Vollzugriff auf den Service zu. IAM-identitätsbasierte Richtlinien oder servicespezifische Richtlinien werden von einer Endpunktrichtlinie nicht überschrieben oder ersetzt. Endpunktrichtlinien steuern unabhängig vom Endpunkt den Zugriff auf den angegebenen Service.

Weitere Informationen finden Sie unter Steuerung des Zugriffs auf Services mit VPC-Endpunkten im Amazon VPC User Guide.

Im Folgenden finden Sie ein Beispiel für eine Endpunktrichtlinie für Amazon Managed Service für Prometheus. Diese Richtlinie ermöglicht Benutzern mit der Rolle `PromUser`, die über die VPC eine Verbindung zu Amazon Managed Service für Prometheus herstellen, Workspaces und Regelgruppen anzuzeigen, aber beispielsweise keine Workspaces zu erstellen oder zu löschen.

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"
                ]
            }
        }
    ]
}
```

Das folgende Beispiel zeigt eine Richtlinie, die nur erlaubt, dass Anfragen, die von einer bestimmten IP-Adresse in der angegebenen VPC kommen, erfolgreich sind. Anfragen von anderen IP-Adressen schlagen fehl.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": "aps:*",
            "Effect": "Allow",
            "Principal": "*",
            "Resource": "*",
            "Condition": {
                "IpAddress": {
                    "aws:VpcSourceIp": "192.0.2.123"
                },
        "StringEquals": {
                    "aws:SourceVpc": "vpc-0123456789abcdef0"
                }
            }
        }
    ]
}
```

Fehlerbehebung bei Amazon Managed Service for Prometheus-Fehlern

Nutzen Sie die folgenden Abschnitte, um Probleme zu beheben, die Amazon Managed Service für Prometheus aufweist.

Themen

- Fehler 429 oder Limit überschritten
- Ich sehe doppelte Beispiele
- Ich sehe Fehler bei Beispiel-Zeitstempeln
- Mir wird eine Fehlermeldung im Zusammenhang mit einem Limit angezeigt
- Ihre lokale Prometheus-Server-Ausgabe überschreitet das Limit.
- Einige meiner Daten werden nicht angezeigt
- HTTP 422-Fehler: Abfrage-Timeout

Fehler 429 oder Limit überschritten

Wenn Sie einen 429-Fehler ähnlich dem folgenden Beispiel sehen, haben Ihre Anfragen die Erfassungskontingente von Amazon Managed Service für Prometheus überschritten.

```
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
```

Wenn Sie einen 429-Fehler ähnlich dem folgenden Beispiel sehen, haben Ihre Anfragen das Kontingent von Amazon Managed Service für Prometheus für die Anzahl der aktiven Metriken in einem Workspace überschritten.

```
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
```

Wenn Sie einen 429-Fehler ähnlich dem folgenden Beispiel sehen, haben Ihre Anfragen das Amazon Managed Service for Prometheus-Kontingent für die Rate (Transaktionen pro Sekunde) überschritten, mit der Sie Daten über die `RemoteWrite` Prometheus-kompatible API an Ihren Workspace senden können.

```
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\"}"
```

Wenn Sie einen 400-Fehler ähnlich dem folgenden Beispiel sehen, haben Ihre Anfragen das Amazon Managed Service for Prometheus-Kontingent für aktive Zeitreihen überschritten. Einzelheiten darüber, wie aktive Zeitreihenkontingente behandelt werden, finden Sie unter. Standardkontingente für aktive Serien

```
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)"
```

Weitere Informationen zum Servicekontingent von Amazon Managed Service für Prometheus und dazu, wie Sie Erhöhungen beantragen können, finden Sie unter Servicekontingente von Amazon Managed Service für Prometheus

Ich sehe doppelte Beispiele

Wenn Sie eine Prometheus Gruppe mit hoher Verfügbarkeit verwenden, müssen Sie externe Labels auf Ihren Prometheus-Instances verwenden, um die Deduplizierung einzurichten. Weitere Informationen finden Sie unter Deduplizierung von Hochverfügbarkeitsmetriken, die an Amazon Managed Service für Prometheus gesendet werden.

Weitere Probleme im Zusammenhang mit duplizierten Daten werden im nächsten Abschnitt behandelt.

Ich sehe Fehler bei Beispiel-Zeitstempeln

Amazon Managed Service for Prometheus nimmt die Daten der Reihe nach auf und erwartet, dass jede Probe einen Zeitstempel hat, der nach der vorherigen Probe liegt.

Wenn Ihre Daten nicht in der richtigen Reihenfolge ankommen, werden Ihnen Fehler zu`out-of-order samples`, oder angezeigt. `duplicate sample for timestamp` `samples with different value but same timestamp` Diese Probleme werden in der Regel durch eine falsche Einrichtung des Clients verursacht, der Daten an Amazon Managed Service for Prometheus sendet. Wenn Sie einen Prometheus-Client verwenden, der im Agentenmodus ausgeführt wird, überprüfen Sie die Konfiguration auf Regeln mit doppelten Seriennamen oder doppelten Zielen. Wenn Ihre Metriken den Zeitstempel direkt angeben, überprüfen Sie, ob sie nicht in der richtigen Reihenfolge sind.

Du kannst ein Zeitfenster einrichten, in dem dein Workspace nicht in der richtigen Reihenfolge Proben akzeptiert. Weitere Informationen finden Sie unter Konfiguriere deinen Workspace.

Weitere Informationen darüber, wie das funktioniert oder wie Sie Ihr Setup überprüfen können, finden Sie im Blogbeitrag Understanding Duplicate Samples and Out-of-order Timestamp Errors in Prometheus von Prom Labs.

Mir wird eine Fehlermeldung im Zusammenhang mit einem Limit angezeigt

> **Note**
>
> Amazon Managed Service for Prometheus bietet CloudWatch Nutzungsmetriken zur Überwachung der Prometheus-Ressourcennutzung. Mithilfe der Alarmfunktion für

CloudWatch Nutzungsmetriken können Sie die Ressourcen und die Nutzung von Prometheus überwachen, um Limitfehler zu vermeiden.

Wenn Sie eine der folgenden Fehlermeldungen sehen, können Sie eine Erhöhung eines der Kontingente von Amazon Managed Service für Prometheus beantragen, um das Problem zu lösen. Weitere Informationen finden Sie unter Servicekontingente von Amazon Managed Service für Prometheus.

- Das Serienlimit pro Benutzer `<value>` wurde überschritten. Bitte wenden Sie sich an den Administrator, um es zu erhöhen
- Das Limit pro metrischer Datenreihe `<value>` wurde überschritten. Bitte wenden Sie sich an den Administrator, um es zu erhöhen
- Limit für die Erfassungsrate (...) überschritten
- Serie: Serie hat zu viele Etiketten (...): '%s'
- der Abfragezeitbereich überschreitet das Limit (Abfragelänge: xxx, Limit: yyy)
- Die Abfrage hat beim Erfassen von Chunks von Ingestern die maximale Anzahl von Chunks erreicht
- Limit überschritten. Maximale Workspaces pro Konto.

Ihre lokale Prometheus-Server-Ausgabe überschreitet das Limit.

Amazon Managed Service für Prometheus verfügt über Servicekontingente für die Datenmenge, die ein Workspace von Prometheus-Servern empfangen kann. Um die Datenmenge zu ermitteln, die Ihr Prometheus-Server an Amazon Managed Service für Prometheus sendet, können Sie die folgenden Abfragen auf Ihrem Prometheus-Server ausführen. Wenn Sie feststellen, dass Ihre Prometheus-Ausgabe ein Limit von Amazon Managed Service für Prometheus überschreitet, können Sie eine Erhöhung des entsprechenden Servicekontingents beantragen. Weitere Informationen finden Sie unter Servicekontingente von Amazon Managed Service für Prometheus.

Abfragen an Ihren lokalen, selbst ausgeführten Prometheus-Server, um die Ausgabelimits zu ermitteln.

Datentyp	Zu verwendende Abfragen	
Aktuelle aktive Serie	`prometheus_tsdb_head_series`	
Aktuelle Erfassungsrate	`rate(prometheus_tsdb_head_samples_appended_total[5m])`	
Most-to-least Liste der aktiven Serien pro Metrikname	`sort_desc(count by(__name__)({__name__!=""}))`	
Anzahl der Etiketten pro metrischer Serie	`group by(mylabelname)({__name__!=""})`	

Einige meiner Daten werden nicht angezeigt

Daten, die an Amazon Managed Service for Prometheus gesendet werden, können aus verschiedenen Gründen verworfen werden. Die folgende Tabelle zeigt Gründe, warum Daten möglicherweise verworfen und nicht aufgenommen werden.

Sie können die Menge und die Gründe, aus denen Daten verworfen wurden, mit Amazon verfolgen. CloudWatch Weitere Informationen finden Sie unter Verwenden Sie CloudWatch Metriken, um die Ressourcen von Amazon Managed Service for Prometheus zu überwachen.

Grund	Bedeutung
greater_than_max_sample_age	Logzeilen, die älter als die aktuelle Uhrzeit sind, werden gelöscht
new-value-for-timestamp	Doppelte Proben werden mit demselben Zeitstempel wie die vorherige Probe, aber mit unterschiedlichen Werten gesendet.
per_metric_series_limit	Der Benutzer hat das Limit für aktive Serien pro Metrik erreicht
per_user_series_limit	Der Benutzer hat das Limit für die Gesamtzahl der aktiven Serien erreicht
rate_limited	Erfassungsrate begrenzt
sample-out-of-order	Die Proben werden nicht in der richtigen Reihenfolge versandt und können nicht bearbeitet werden. Dieser Grund wird nur angezeigt, wenn das Zeitfenster für die Nichtbestellung auf 0 gesetzt ist.
Beispiel ist zu alt	Die Probe ist älter als das konfigurierte Zeitfenster, in dem die Reihenfolge abgelaufen ist, und kann nicht verarbeitet werden.
Ratenlimit bei Nichtbestellung	Out-of-order Die maximale Aufnahmerate wurde erreicht und die Probe kann nicht verarbeitet werden. Weitere Informationen finden Sie unter Servicekontingente von Amazon Managed Service für Prometheus.
label_value_too_long	Der Labelwert ist länger als die zulässige Zeichenbe schränkung
max_label_names_per_series	Der Benutzer hat die Etikettennamen pro Metrik erreicht

Grund	Bedeutung
missing_metric_name	Der Name der Metrik wurde nicht angegeben
metric_name_invalid	Ungültiger Metrikname angegeben
label_invalid	Ungültiges Etikett angegeben
duplicate_label_names	Doppelte Etikettennamen angegeben
native_histogram_sample_size_bytes_exceeded	Die native Histogramm-Stichprobe überschreitet die maximal zulässige Stichprobengröße. Reduzieren Sie die Anzahl der Buckets oder Spans im Histogramm auf der Client-Seite.
native_histogram_invalid_schema	Das native Histogramm hat einen ungültigen Schemawert. Gültige Schemas liegen im Bereich von -4 bis 8. Überprüfen Sie Ihre Prometheus-Client-Konfiguration auf die `NativeHistogramBucketFactor` Einstellung.
native_histogram_invalid	Das native Histogramm schlägt bei der Überprüfung fehl (z. B. negative Bucket-Anzahl, nicht übereinstimmende Bucket-Anzahl oder falsch formatierte Bucket-Spans). Dies deutet in der Regel auf einen Fehler in der Instrumentierungsbibliothek hin.
native_histogram_buckets_exceeded	Das native Histogramm überschreitet die maximale Anzahl an Buckets und die Auflösung kann nicht automatisch reduziert werden. Reduzieren Sie die Anzahl der Buckets auf der Client-Seite mithilfe von. `NativeHistogramMaxBucketNumber`
native_histogram_rate_limited	Eine native Histogrammprobe wurde zurückgewiesen, da die maximale Aufnahmerate für native Histogramme erreicht wurde. Reduzieren Sie die Anzahl der nativen Histogrammreihen oder die Häufigkeit des Scrapes, oder fordern Sie eine Erhöhung des Grenzwerts an.

Grund	Bedeutung
per_user_native_histogram_series_limit	Der Benutzer hat das Limit für aktive Serien für native Histogramme pro Workspace erreicht. Reduzieren Sie die Anzahl der nativen Histogrammreihen oder fordern Sie eine Erhöhung des Limits an.

HTTP 422-Fehler: Abfrage-Timeout

Wenn Ihre Abfrage länger als 1 Minute läuft, gibt Amazon Managed Service for Prometheus einen HTTP 422-Statuscode zurück. Das folgende Beispiel zeigt diesen Fehler:

```
{"status":"error","errorType":"execution","error":"query timed out - consider
 simplifying your query"}
```

Verwenden Sie einen oder mehrere der folgenden Ansätze, um die Evaluierungszeit zu verkürzen:

- Geben Sie ein kürzeres Zeitfenster an, um weniger Proben auszuwerten.
- Fügen Sie Label-Matcher hinzu, um die Anzahl der Serien zu reduzieren, die Ihre Abfrage auswertet.
- Verwenden Sie Aufzeichnungsregeln, um teure Aggregationen, die Sie häufig abfragen, vorab zu berechnen.
- Vermeiden Sie Regex-Matcher bei Labels mit hoher Kardinalität (Beschriftungen mit vielen Einzelwerten).

Weitere Informationen finden Sie unter Fragen Sie Ihre Metriken ab.

Tagging in Amazon Managed Service für Prometheus

Ein Tag ist eine benutzerdefinierte Attributbezeichnung, die Sie einer Ressource AWS zuweisen oder zuweisen. AWS Jedes AWS Tag besteht aus zwei Teilen:

- einem Tag-Schlüssel (z. B. `CostCenter`, `Environment`, `Project` oder `Secret`). Bei Tag-Schlüsseln wird zwischen Groß- und Kleinschreibung unterschieden.
- einem optionalen Feld, das als Tag-Wert bezeichnet wird (z. B. `111122223333`, `Production` oder ein Team-Name). Ein nicht angegebener Tag-Wert entspricht einer leeren Zeichenfolge. Wie bei Tag-Schlüsseln wird auch bei Tag-Werten zwischen Groß- und Kleinschreibung unterschieden.

Zusammen werden sie als Schlüssel-Wert-Paare bezeichnet. Sie können jedem Workspace bis zu 50 Tags zuweisen.

Mithilfe von Tags können Sie Ihre AWS Ressourcen identifizieren und organisieren. Viele AWS Dienste unterstützen Tagging, sodass Sie Ressourcen aus verschiedenen Diensten dasselbe Tag zuweisen können, um anzuzeigen, dass die Ressourcen miteinander verknüpft sind. Sie können zum Beispiel einem Workspace in Amazon Managed Service für Prometheus denselben Tag zuweisen, den Sie auch einem Amazon-S3-Bucket zuweisen. Weitere Informationen zu Markierungsstrategien finden Sie unter Markieren von AWS -Ressourcen.

In Amazon Managed Service für Prometheus können Namespaces sowohl für Workspaces als auch für Regelgruppen mit Tags versehen werden. Sie können die Konsole, oder, verwenden AWS CLI APIs, SDKs um Tags für diese Ressourcen hinzuzufügen, zu verwalten und zu entfernen. Sie können nicht nur Ihre Workspaces und Regelgruppen-Namespaces mit Tags identifizieren, organisieren und verfolgen, sondern auch Tags in IAM-Richtlinien verwenden, um zu kontrollieren, wer Ihre Ressourcen für Amazon Managed Service für Prometheus anzeigen und mit ihnen interagieren kann.

Tag-Einschränkungen

Die folgenden grundlegenden Einschränkungen gelten für Tags (Markierungen):

- Jede Ressource kann maximal 50 Tags haben.
- Jeder Tag muss für jede Ressource eindeutig sein. Jeder Tag kann nur einen Wert haben.
- Die maximale Länge des Tag-Schlüssels beträgt 128 Unicode-Zeichen in UTF-8.
- Die maximale Länge des Tag-Wertes beträgt 256 Unicode-Zeichen in UTF-8.

- Wenn Ihr Tagging-Schema für mehrere AWS Dienste und Ressourcen verwendet wird, denken Sie daran, dass für andere Dienste möglicherweise Einschränkungen hinsichtlich der zulässigen Zeichen gelten. Allgemein erlaubte Zeichen sind Buchstaben, Ziffern und Leerzeichen, die in UTF-8 darstellbar sind, sowie die folgenden Zeichen: . : + = @ _ / - (Bindestrich).
- Bei Tag-Schlüsseln und -Werten muss die Groß- und Kleinschreibung beachtet werden. Eine bewährte Methode besteht darin, sich für eine einheitliche Schreibweise der Tag-Benennungen zu entscheiden und diese Strategie für alle Ressourcentypen umzusetzen. Entscheiden Sie sich beispielsweise für `Costcenter`, `costcenter` oder `CostCenter` und verwenden Sie diese Konvention für alle Tags. Vermeiden Sie die Verwendung von ähnlichen Tags mit uneinheitlicher Fallunterscheidung.
- Verwenden Sie weder `aws:` noch `AWS:` oder Kombinationen aus Groß- und Kleinbuchstaben von diesen als Präfix für Schlüssel oder Werte, da sie für die -Verwendung reserviert sind. Diese sind nur für die AWS Verwendung reserviert. Sie können keine Tag-Schlüssel oder -Werte mit diesem Präfix bearbeiten oder löschen. Tags mit diesem Präfix werden nicht auf Ihr tags-per-resource Limit angerechnet.

Themen

- Schlagwort: Amazon Managed Service für Prometheus-Workspaces
- Namespaces von Regelgruppen markieren

Schlagwort: Amazon Managed Service für Prometheus-Workspaces

Tags sind benutzerdefinierte Labels, die einer Ressource zugewiesen werden können. Sie enthalten einen eindeutigen Schlüssel und einen optionalen Wert (in einem Schlüssel-Wert-Paar). Tags helfen Ihnen, Ihre AWS -Ressourcen zu identifizieren und zu organisieren. In Amazon Managed Service for Prometheus können Arbeitsbereiche (und Regelgruppen-Namespaces) mit Tags versehen werden. Sie können die Konsole, die AWS CLI oder verwenden APIs, SDKs um Tags für diese Ressourcen hinzuzufügen, zu verwalten und zu entfernen. Zusätzlich zur Identifizierung, Organisation und Nachverfolgung Ihrer Workspaces mit Tags können Sie Tags in IAM-Richtlinien verwenden, um zu kontrollieren, wer Ihre Amazon Managed Service for Prometheus-Ressourcen ansehen und mit ihnen interagieren kann.

Verwenden Sie die Verfahren in diesem Abschnitt, um mit Tags für Workspaces von Amazon Managed Service für Prometheus zu arbeiten.

Themen

- Einem Workspace ein Tag hinzufügen
- Tags für einen Workspace anzeigen
- Bearbeiten von Tags für einen Workspace
- Ein Tag aus einem Workspace entfernen

Einem Workspace ein Tag hinzufügen

Das Hinzufügen von Tags zu einem Workspace in Amazon Managed Service für Prometheus kann Ihnen helfen, Ihre AWS -Ressourcen zu identifizieren und zu organisieren und den Zugriff auf sie zu verwalten. Fügen Sie zunächst ein oder mehrere Tags (Schlüssel-Wert-Paare) zu einem Workspace hinzu. Nach der Erstellung der Tags können Sie IAM-Richtlinien für die Verwaltung des Zugriffs auf den Workspace basierend auf diesen Tags erstellen. Sie können die Konsole oder die verwenden, um Tags AWS CLI zu einem Amazon Managed Service for Prometheus Workspace hinzuzufügen.

> **Important**
>
> Das Hinzufügen von Tags zu einem Workspace kann sich auf den Zugriff auf diesen Workspace auswirken. Bevor Sie einen Tag zu einem Workspace hinzufügen, müssen Sie alle IAM-Richtlinien überprüfen, die möglicherweise Tags für die Steuerung des Zugriffs auf Ressourcen verwenden.

Weitere Informationen zum Hinzufügen von Tags zu einem Workspace in Amazon Managed Service für Prometheus während dessen Erstellung finden Sie unter Erstellen Sie einen Amazon Managed Service für Prometheus Workspace.

Themen

- Einem Workspace (Konsole) einen Tag hinzufügen
- Einem Workspace einen Tag hinzufügen (AWS CLI)

Einem Workspace (Konsole) einen Tag hinzufügen

Sie können mit der Konsole einen oder mehrere Tags zu einem Workspace in Amazon Managed Service für Prometheus hinzufügen.

1. Öffnen Sie die Amazon Managed Service for Prometheus-Konsole unter. https://console.aws.amazon.com/prometheus/
2. Wählen Sie im Navigationsbereich das Menüsymbol.
3. Wählen Sie Alle Workspaces.
4. Wählen Sie die Workspace-ID des Workspace, den Sie verwalten möchten.
5. Wählen Sie die Registerkarte Tags aus.
6. Wenn dem Workspace in Amazon Managed Service für Prometheus keine Tags hinzugefügt wurden, wählen Sie Tag erstellen. Andernfalls wählen Sie Tags verwalten.
7. Geben Sie für Schlüssel einen Namen für das Tag ein. Sie können einen optionalen Wert für das Tag unter Wert hinzufügen.
8. (Optional) Zum Hinzufügen eines weiteren Tags wählen Sie Tag hinzufügen erneut aus.
9. Wenn Sie fertig mit dem Hinzufügen der Tags sind, klicken Sie auf Änderungen speichern.

Einem Workspace einen Tag hinzufügen (AWS CLI)

Gehen Sie wie folgt vor, AWS CLI um ein Tag zu einem Amazon Managed Service for Prometheus Workspace hinzuzufügen. Informationen darüber, wie Sie beim Erstellen einem Workspace einen Tag hinzufügen können, finden Sie unter Erstellen Sie einen Amazon Managed Service für Prometheus Workspace.

In diesen Schritten gehen wir davon aus, dass Sie bereits eine aktuelle Version von installiert AWS CLI oder auf die aktuelle Version aktualisiert haben. Weitere Informationen finden Sie unter Installieren der AWS Command Line Interface.

Führen Sie am Terminal oder über die Befehlszeile den Befehl tag-resource aus und geben Sie dabei den Amazon-Ressourcennamen (ARN) des Workspace an, für den Sie Tags hinzufügen möchten, sowie den Schlüssel und Wert des hinzuzufügenden Tags. Sie können einem Workspace mehrere Tags hinzufügen. Um beispielsweise einen Amazon Managed Service for Prometheus Workspace namens My-Workspace mit zwei Tags zu taggen, einem Tag-Schlüssel *Status* mit dem Tag-Wert von *Secret* und einem Tag-Schlüssel *Team* mit dem Tag-Wert von: *My-Team*

```
aws amp tag-resource --resource-arn arn:aws:aps:us-
west-2:123456789012:workspaces/IDstring
--tags Status=Secret,Team=My-Team
```

Bei erfolgreicher Ausführung gibt dieser Befehl nichts zurück.

Tags für einen Workspace anzeigen

Mithilfe von Stichwörtern können Sie Ihre AWS Ressourcen identifizieren und organisieren und den Zugriff darauf verwalten. Weitere Informationen zu Tagging-Strategien finden Sie unter Ressourcen taggen AWS.

Anzeigen von Tags für einen Workspace in Amazon Managed Service für Prometheus (Konsole)

Sie können die Konsole verwenden, um die Tags anzuzeigen, die einem Workspace in Amazon Managed Service für Prometheus zugeordnet sind.

1. Öffnen Sie die Amazon Managed Service for Prometheus-Konsole unter. https://console.aws.amazon.com/prometheus/
2. Wählen Sie im Navigationsbereich das Menüsymbol.
3. Wählen Sie Alle Workspaces.
4. Wählen Sie die Workspace-ID des Workspace, den Sie verwalten möchten.
5. Wählen Sie die Registerkarte Tags aus.

Anzeigen von Tags für einen Workspace in Amazon Managed Service für Prometheus (AWS CLI)

Gehen Sie wie folgt vor, AWS CLI um die AWS Tags für einen Workspace anzuzeigen. Wenn keine Tags hinzugefügt wurden, ist die zurückgegebene Liste leer.

Führen Sie am Terminal oder über die Befehlszeile den Befehl list-tags-for-resource aus. Um beispielsweise eine Liste der Tag-Schlüssel und Tag-Werte für einen Workspace anzuzeigen:

```
aws amp list-tags-for-resource --resource-arn arn:aws:aps:us-
west-2:123456789012:workspace/IDstring
```

Bei erfolgreicher Ausführung gibt dieser Befehl etwa wie folgt aussehende Informationen zurück:

```
{
    "tags": {
        "Status": "Secret",
        "Team": "My-Team"
    }
```

```
}
```

Bearbeiten von Tags für einen Workspace

Sie können den Wert für ein Tag ändern, das mit einem Projekt verknüpft ist. Sie können auch den Namen des Schlüssels ändern. Dies entspricht dem Entfernen des aktuellen Tags und dem Hinzufügen eines anderen Tags mit dem neuen Namen und demselben Wert wie der andere Schlüssel.

> **Important**
>
> Das Bearbeiten von Tags für einen Workspace in Amazon Managed Service für Prometheus kann sich auf den Zugriff auf diese Workspaces auswirken. Bevor Sie den Namen (Schlüssel) oder den Wert eines Tags für einen Workspace bearbeiten, müssen Sie alle IAM-Richtlinien überprüfen, die den Schlüssel oder Wert eines Tags zum Steuern des Zugriffs auf Ressourcen (beispielsweise Repositorys) verwenden.

Ein Tag für einen Workspace in Amazon Managed Service für Prometheus bearbeiten (Konsole)

Sie können mit der Konsole die Tags bearbeiten, die einem Workspace in Amazon Managed Service für Prometheus zugeordnet sind.

1. Öffnen Sie die Amazon Managed Service for Prometheus-Konsole unter. https://console.aws.amazon.com/prometheus/
2. Wählen Sie im Navigationsbereich das Menüsymbol.
3. Wählen Sie Alle Workspaces.
4. Wählen Sie die Workspace-ID des Workspace, den Sie verwalten möchten.
5. Wählen Sie die Registerkarte Tags aus.
6. Wenn dem Workspace noch keine Tags hinzugefügt wurden, wählen Sie Tag hinzufügen aus. Andernfalls wählen Sie Tags verwalten.
7. Geben Sie für Schlüssel einen Namen für das Tag ein. Sie können einen optionalen Wert für das Tag unter Wert hinzufügen.
8. (Optional) Zum Hinzufügen eines weiteren Tags wählen Sie Tag hinzufügen erneut aus.
9. Wenn Sie fertig mit dem Hinzufügen der Tags sind, klicken Sie auf Änderungen speichern.

Bearbeiten von Tags für einen Workspace in Amazon Managed Service für Prometheus (AWS CLI)

Gehen Sie wie folgt vor, um mit dem ein Tag für einen Workspace AWS CLI zu aktualisieren. Sie können den Wert für einen vorhandenen Schlüssel ändern oder einen anderen Schlüssel hinzufügen.

Führen Sie im Terminal oder in der Befehlszeile den Befehl tag-resource aus und bestimmen Sie dabei den Amazon-Ressourcennamen (ARN) des Workspace in Amazon Managed Service für Prometheus, für den Sie ein Tag aktualisieren möchten, sowie den Schlüssel und den Wert des Tags, das Sie aktualisieren möchten:

```
aws amp tag-resource --resource-arn arn:aws:aps:us-
west-2:123456789012:workspace/IDstring --tags Team=New-Team
```

Ein Tag aus einem Workspace entfernen

Sie können ein oder mehrere mit einem Workspace verknüpfte Tags entfernen. Durch das Entfernen eines Tags wird das Tag nicht aus anderen AWS Ressourcen gelöscht, die mit diesem Tag verknüpft sind.

> **Important**
>
> Das Entfernen von Tags für einen Workspace in Amazon Managed Service für Prometheus kann sich auf den Zugriff auf diesen Workspaces auswirken. Bevor Sie ein Tag aus einem Workspace entfernen, müssen Sie alle IAM-Richtlinien überprüfen, die den Schlüssel oder Wert eines Tags zum Steuern des Zugriffs auf Ressourcen (beispielsweise Repositorys) verwenden.

Ein Tag aus einem Workspace in Amazon Managed Service für Prometheus (Konsole) entfernen

Sie können die Konsole verwenden, um die Verknüpfung zwischen einem Tag und einem Workspace zu entfernen.

1. Öffnen Sie die Amazon Managed Service for Prometheus-Konsole unter. https://console.aws.amazon.com/prometheus/
2. Wählen Sie im Navigationsbereich das Menüsymbol.

3. Wählen Sie Alle Workspaces.
4. Wählen Sie die Workspace-ID des Workspace, den Sie verwalten möchten.
5. Wählen Sie die Registerkarte Tags aus.
6. Wählen Sie Tags verwalten aus.
7. Finden Sie das Tag, das Sie entfernen möchten und wählen Sie die Option Entfernen aus.

Ein Tag aus einem Workspace in Amazon Managed Service für Prometheus entfernen (AWS CLI)

Gehen Sie wie folgt vor, um mit dem ein Tag aus einem Workspace AWS CLI zu entfernen. Durch das Entfernen wird ein Tag nicht gelöscht, sondern lediglich die Verknüpfung zwischen dem Tag und der Workspace entfernt.

> **Note**
>
> Wenn Sie einen Workspace in Amazon Managed Service für Prometheus löschen, werden alle Tag-Zuordnungen aus dem gelöschten Workspace entfernt. Sie müssen keine Tags entfernen, bevor Sie einen Workspace löschen.

Führen Sie am Terminal oder über die Befehlszeile den Befehl untag-resource aus und geben Sie dabei den Amazon-Ressourcennamen (ARN) des Workspace an, für den Sie Tags hinzufügen möchten, sowie den Tag-Schlüssel des zu entfernenden Tags. Um beispielsweise ein Tag in einem Workspace namens My-Workspace mit dem Tag-Schlüssel *Status* zu entfernen:

```
aws amp untag-resource --resource-arn arn:aws:aps:us-
west-2:123456789012:workspace/IDstring --tag-keys Status
```

Bei erfolgreicher Ausführung gibt dieser Befehl nichts zurück. Um die dem Workspace zugeordneten Tags zu überprüfen, führen Sie den Befehl list-tags-for-resource aus.

Namespaces von Regelgruppen markieren

Tags sind benutzerdefinierte Bezeichnungen, die einer Ressource zugewiesen werden können. Sie enthalten einen eindeutigen Schlüssel und einen optionalen Wert (in einem Schlüssel-Wert-Paar). Tags helfen Ihnen, Ihre AWS -Ressourcen zu identifizieren und zu organisieren. In Amazon Managed Service for Prometheus können die Namespaces (und Workspaces) für Regelgruppen

mit Tags versehen werden. Sie können die Konsole, die AWS CLI oder verwenden APIs, SDKs um Tags für diese Ressourcen hinzuzufügen, zu verwalten und zu entfernen. Neben der Identifizierung, Organisation und Nachverfolgung Ihrer Regelgruppen-Namespaces mit Tags können Sie Tags in IAM-Richtlinien verwenden, um zu kontrollieren, wer Ihre Amazon Managed Service for Prometheus-Ressourcen ansehen und mit ihnen interagieren kann.

Verwenden Sie die Verfahren in diesem Abschnitt, um mit Tags für die Regelgruppen-Namespaces von Amazon Managed Service für Prometheus zu arbeiten.

Themen

- Hinzufügen eines Tags zum Regelgruppen-Namespace
- Anzeigen von Tags für einen Regelgruppen-Namespace
- Bearbeiten von Tags für einen Regelgruppen-Namespace
- Entfernen Sie ein Tag aus einem Regelgruppen-Namespace

Hinzufügen eines Tags zum Regelgruppen-Namespace

Das Hinzufügen von Tags zu den Namespaces für Regelgruppen von Amazon Managed Service für Prometheus kann Ihnen helfen, Ihre AWS Ressourcen zu identifizieren und zu organisieren und den Zugriff darauf zu verwalten. Fügen Sie zunächst ein oder mehrere Tags (Schlüssel-Wert-Paare) zu einem Regelgruppen-Namespace hinzu. Nach der Erstellung der Tags können Sie IAM-Richtlinien für die Verwaltung des Zugriffs auf den Namespace basierend auf diesen Tags erstellen. Sie können die Konsole oder den verwenden, um Tags AWS CLI zu einem Amazon Managed Service for Prometheus Regelgruppen-Namespace hinzuzufügen.

> **Important**
>
> Das Hinzufügen von Tags zu einem Regelgruppen-Namespace kann sich auf den Zugriff auf den Namespace dieser Regelgruppe auswirken. Bevor Sie ein Tag hinzufügen, müssen Sie alle IAM-Richtlinien überprüfen, die möglicherweise Tags für die Steuerung des Zugriffs auf Ressourcen verwenden.

Weitere Informationen zum Hinzufügen von Tags zu einem Regelgruppen-Namespace während ihrer Erstellung finden Sie unter Erstellen Sie eine Regeldatei.

Themen

- Ein Tag zum Namespace einer Regelgruppe hinzufügen (Konsole)
- Dem Namespace einer Regelgruppe ein Tag hinzufügen (AWS CLI)

Ein Tag zum Namespace einer Regelgruppe hinzufügen (Konsole)

Sie können die Konsole verwenden, um einem Regelgruppen-Namespace von Amazon Managed Service für Prometheus ein oder mehrere Tags hinzuzufügen.

1. Öffnen Sie die Amazon Managed Service for Prometheus-Konsole unter. https://console.aws.amazon.com/prometheus/
2. Wählen Sie im Navigationsbereich das Menüsymbol.
3. Wählen Sie Alle Workspaces.
4. Wählen Sie die Workspace-ID des Workspace, den Sie verwalten möchten.
5. Wählen Sie die Registerkarte Regelverwaltung.
6. Klicken Sie auf die Schaltfläche neben dem Namen des Namespace und wählen Sie Bearbeiten.
7. Wählen Sie Tags erstellen, Neues Tag hinzufügen.
8. Geben Sie für Schlüssel einen Namen für das Tag ein. Sie können einen optionalen Wert für das Tag unter Wert hinzufügen.
9. (Optional) Zum Hinzufügen eines weiteren Tags wählen Sie Neues Tag hinzufügen erneut aus.
10. Wenn Sie fertig mit dem Hinzufügen der Tags sind, klicken Sie auf Änderungen speichern.

Dem Namespace einer Regelgruppe ein Tag hinzufügen (AWS CLI)

Gehen Sie wie folgt vor, um den AWS CLI Namespace To add a tag to an Amazon Managed Service for Prometheus Regelgruppen zu verwenden. Informationen zum Hinzufügen eines Tags beim Erstellen eines Regelgruppen-Namespace finden Sie unter Laden Sie eine Regelkonfigurationsdatei auf Amazon Managed Service for Prometheus hoch.

In diesen Schritten gehen wir davon aus, dass Sie bereits eine aktuelle Version von installiert AWS CLI oder auf die aktuelle Version aktualisiert haben. Weitere Informationen finden Sie unter Installieren der AWS Command Line Interface.

Führen Sie am Terminal oder in der Befehlszeile den Befehl tag-resource aus und geben Sie dabei den Amazon-Ressourcennamen (ARN) des Regelgruppen-Namespace an, für die Sie Tags hinzufügen möchten, sowie den Schlüssel und Wert des hinzuzufügenden Tags. Sie können einem

Regelgruppen-Namespace mehrere Tags hinzufügen. Um beispielsweise einen Amazon Managed Service for Prometheus Namespace namens My-Workspace mit zwei Tags zu taggen, einem Tag-Schlüssel *Status* mit dem Tag-Wert von *Secret* und einem Tag-Schlüssel *Team* mit dem Tag-Wert von: *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
```

Bei erfolgreicher Ausführung gibt dieser Befehl nichts zurück.

Anzeigen von Tags für einen Regelgruppen-Namespace

Mithilfe von Tags können Sie Ihre AWS Ressourcen identifizieren und organisieren und den Zugriff darauf verwalten. Weitere Informationen zu Tagging-Strategien finden Sie unter Ressourcen taggen AWS.

Tags für einen Regelgruppen-Namespace von Amazon Managed Service für Prometheus anzeigen (Konsole)

Sie können mit der Konsole die Tags anzeigen, die einem Namespace von Amazon Managed Service für Prometheus zugeordnet sind.

1. Öffnen Sie die Amazon Managed Service for Prometheus-Konsole unter. https://console.aws.amazon.com/prometheus/
2. Wählen Sie im Navigationsbereich das Menüsymbol.
3. Wählen Sie Alle Workspaces.
4. Wählen Sie die Workspace-ID des Workspace, den Sie verwalten möchten.
5. Wählen Sie die Registerkarte Regelverwaltung.
6. Wählen Sie den Namespace aus.

Anzeigen von Tags für einen Workspace in Amazon Managed Service für Prometheus (AWS CLI)

Gehen Sie wie folgt vor, um den AWS CLI zum Anzeigen der AWS Tags für einen Regelgruppen-Namespace zu verwenden. Wenn keine Tags hinzugefügt wurden, ist die zurückgegebene Liste leer.

Führen Sie am Terminal oder über die Befehlszeile den Befehl list-tags-for-resource aus. So zeigen Sie beispielsweise eine Liste von Tag-Schlüsseln und Tag-Werten für einen Regelgruppen-Namespace an:

```
aws amp list-tags-for-resource --resource-arn rn:aws:aps:us-
west-2:123456789012:rulegroupsnamespace/IDstring/namespace_name
```

Bei erfolgreicher Ausführung gibt dieser Befehl etwa wie folgt aussehende Informationen zurück:

```
{
    "tags": {
        "Status": "Secret",
        "Team": "My-Team"
    }
}
```

Bearbeiten von Tags für einen Regelgruppen-Namespace

Sie können den Wert für ein Tag ändern, das mit einem Regelgruppen-Namespace verknüpft ist. Sie können auch den Namen des Schlüssels ändern. Dies entspricht dem Entfernen des aktuellen Tags und dem Hinzufügen eines anderen Tags mit dem neuen Namen und demselben Wert wie der andere Schlüssel.

> **Important**
>
> Das Bearbeiten von Tags für einen Regelgruppen-Namespace kann sich auf den Zugriff darauf auswirken. Bevor Sie den Namen (Schlüssel) oder den Wert eines Tags für eine Ressource bearbeiten, müssen Sie alle IAM-Richtlinien überprüfen, die den Schlüssel oder Wert eines Tags möglicherweise für die Steuerung des Zugriffs auf Ressourcen verwenden.

Ein Tag für einen Regelgruppen-Namespace von Amazon Managed Service für Prometheus bearbeiten (Konsole)

Sie können mit der Konsole die Tags bearbeiten, die einem Regelgruppen-Namespace von Amazon Managed Service für Prometheus zugeordnet sind.

1. Öffnen Sie die Amazon Managed Service for Prometheus-Konsole unter. https://console.aws.amazon.com/prometheus/

2. Wählen Sie im Navigationsbereich das Menüsymbol.
3. Wählen Sie Alle Workspaces.
4. Wählen Sie die Workspace-ID des Workspace, den Sie verwalten möchten.
5. Wählen Sie die Registerkarte Regelverwaltung.
6. Wählen Sie den Namen des Namespace aus.
7. Wählen Sie Tags Verwalten und Neues Tag hinzufügen.
8. Um den Wert eines vorhandenen Tags zu ändern, geben Sie den neuen Wert für Wert ein.
9. Wenn Sie einen zusätzlichen Tag hinzufügen möchten, wählen Sie Neues Tag hinzufügen aus.
10. Wenn Sie fertig mit dem Hinzufügen und Bearbeiten der Tags sind, klicken Sie auf Änderungen speichern.

Tags für einen Regelgruppen-Namespace von Amazon Managed Service für Prometheus bearbeiten (AWS CLI)

Gehen Sie wie folgt vor, um den AWS CLI Namespace zum Aktualisieren eines Tags für eine Regelgruppe zu verwenden. Sie können den Wert für einen vorhandenen Schlüssel ändern oder einen anderen Schlüssel hinzufügen.

Führen Sie im Terminal oder in der Befehlszeile den Befehl tag-resource aus und geben Sie dabei den Amazon-Ressourcennamen (ARN) der Ressource, für die Sie ein Tag aktualisieren möchten, sowie den Schlüssel und den Wert des Tags an:

```
aws amp tag-resource --resource-arn rn:aws:aps:us-
west-2:123456789012:rulegroupsnamespace/IDstring/namespace_name --tags Team=New-Team
```

Entfernen Sie ein Tag aus einem Regelgruppen-Namespace

Sie können ein oder mehrere Tags entfernen, die mit einem Regelgruppen-Namespace verknüpft sind. Durch das Entfernen eines Tags wird das Tag nicht aus anderen AWS Ressourcen gelöscht, die mit diesem Tag verknüpft sind.

> **Important**
>
> Das Entfernen von Tags für eine Ressource kann sich auf den Zugriff auf diese Ressource auswirken. Bevor Sie ein Tag aus einer Ressource entfernen, müssen Sie alle IAM-

Richtlinien überprüfen, die den Schlüssel oder Wert eines Tags zum Steuern des Zugriffs auf Ressourcen (beispielsweise Repositorys) verwenden.

Ein Tag aus einem Regelgruppen-Namespace von Amazon Managed Service für Prometheus entfernen (Konsole)

Sie können die Konsole verwenden, um die Verknüpfung zwischen einem Tag und einem Regelgruppen-Namespace zu entfernen.

1. Öffnen Sie die Amazon Managed Service for Prometheus-Konsole unter. https://console.aws.amazon.com/prometheus/
2. Wählen Sie im Navigationsbereich das Menüsymbol.
3. Wählen Sie Alle Workspaces.
4. Wählen Sie die Workspace-ID des Workspace, den Sie verwalten möchten.
5. Wählen Sie die Registerkarte Regelverwaltung.
6. Wählen Sie den Namen des Namespace aus.
7. Wählen Sie Tags verwalten aus.
8. Wählen Sie neben dem Tag, das Sie entfernen möchten, die Option Remove (Entfernen) aus.
9. Klicken Sie auf Save changes (Änderungen speichern), sobald Sie fertig sind.

Ein Tag aus einem Regelgruppen-Namespace von Amazon Managed Service für Prometheus entfernen (AWS CLI)

Gehen Sie wie folgt vor, um mit dem ein Tag aus dem Namespace einer Regelgruppe AWS CLI zu entfernen. Durch das Entfernen wird ein Tag nicht gelöscht, sondern lediglich die Verknüpfung zwischen dem Tag und dem Regelgruppen-Namespace entfernt.

> **Note**
>
> Wenn Sie einen Regelgruppen-Namespace von Amazon Managed Service für Prometheus löschen, werden alle Tag-Zuordnungen aus dem gelöschten Namespace entfernt. Sie müssen keine Tags entfernen, bevor Sie einen Namespace löschen.

Führen Sie am Terminal oder über die Befehlszeile den Befehl untag-resource aus und geben Sie dabei den Amazon-Ressourcennamen (ARN) des Regelgruppen-Namespace an, für den Sie Tags hinzufügen möchten, sowie den Tag-Schlüssel des zu entfernenden Tags. Um beispielsweise ein Tag in einem Arbeitsbereich namens My-Workspace mit dem Tag-Schlüssel zu entfernen: *Status*

```
aws amp untag-resource --resource-arn rn:aws:aps:us-
west-2:123456789012:rulegroupsnamespace/IDstring/namespace_name --tag-keys Status
```

Bei erfolgreicher Ausführung gibt dieser Befehl nichts zurück. Um die der Ressource zugeordneten Tags zu überprüfen, führen Sie den Befehl list-tags-for-resource aus.

Servicekontingente von Amazon Managed Service für Prometheus

In den folgenden beiden Abschnitten werden die mit Amazon Managed Service for Prometheus verbundenen Kontingente und Limits beschrieben.

Servicekontingente

Amazon Managed Service for Prometheus hat die folgenden Kontingente. Amazon Managed Service for Prometheus bietet CloudWatch Nutzungsmetriken zur Überwachung der Prometheus-Ressourcennutzung. Mithilfe der Alarmfunktion für CloudWatch Amazon-Nutzungsmetriken können Sie die Ressourcen und die Nutzung von Prometheus überwachen, um Limitfehler zu vermeiden.

Wenn Ihre Projekte und Arbeitsbereiche wachsen, sollten Sie folgende Kontingente am häufigsten überwachen oder deren Erhöhung beantragen: Aktive Serien pro Arbeitsbereich und Aufnahmerate pro Arbeitsbereich.

Für alle anpassbaren Kontingente kannst du eine Erhöhung des Kontingents beantragen, indem du auf den Link in der Spalte „Anpassbar“ klickst, oder indem du eine Erhöhung des Kontingents beantragst.

Das Limit für aktive Serien pro Workspace wird dynamisch angewendet. Weitere Informationen finden Sie unter Standardkontingente für aktive Serien. Die Aufnahmerate pro Workspace-Kontingent bestimmt, wie schnell du Daten in deinen Workspace aufnehmen kannst. Weitere Informationen finden Sie unter Drosselung der Aufnahme.

> **Note**
>
> Sofern nicht anders angegeben, gelten diese Kontingente pro Workspace. Der Höchstwert für aktive Serien pro Workspace liegt bei 1,5 Milliarden.

Name	Standard	Anpas	Description
Aktive Metriken mit Metadaten pro Workspace	Jede unterstützte Region: 20 000	Nein	Die Anzahl einmalige r aktiver Metriken

Name	Standard	Anpas	Description
			mit Metadaten pro Workspace. Hinweis: Wenn das Limit erreicht ist, wird eine Metrikprobe aufgezeichnet, Metadaten, die den Grenzwert überschreiten, werden jedoch gelöscht.
Aktive Serien pro Workspace	Jede unterstützte Region: 50 000 000	Ja	Die Anzahl der einzelnen aktiven Serien pro Workspace (bis zu einem Maximum von 1,5 Milliarden). Eine Serie ist aktiv, wenn in den letzten 2 Stunden ein Beispiel gemeldet wurde. Die Kapazität von 2 M bis 50 M wird automatisch auf der Grundlage der Nutzung der letzten 30 Minuten angepasst.
Größe der Alarmaggregationsgruppe in der Alert-Manager-Definitionsdatei	Jede unterstützte Region: 1 000	Ja	Die maximale Größe einer Alarmaggregationsgruppe in der Alert-Manager-Definitionsdatei. Jede Labelwertkombination von group_by würde eine Aggregationsgruppe erstellen.

Name	Standard	Anpas	Description
Größe der Definitionsdatei des Alert Managers	Jede unterstützte Region: 1 000 000	Nein	Die maximale Größe einer Warnungs-Manager-Definitionsdatei in Byte.
Größe der Warnungs-Payload im Alert Manager	Jede unterstützte Region: 20 000 000	Nein	Die maximale Größe der Warnungsnutzlast aller Alert Manager-Warnungen pro Arbeitsbereich in Byte. Die Alarmgröße hängt von den Bezeichnungen und Anmerkungen ab.
Warnungen im Alert Manager	Jede unterstützte Region: 1 000	Ja	Die maximale Anzahl gleichzeitiger Alert Manager-Benachrichtigungen pro Arbeitsbereich.
HA-Tracker-Cluster	Jede unterstützte Region: 500	Nein	Die maximale Anzahl von Clustern, die der HA-Tracker für erfasste Beispiel pro Workspace verfolgt.
Erfassungsrate pro Workspace	Jede unterstützte Region: 1.666.666	Ja	Metrische Beispiel-Erfassungsrate pro Workspace pro Sekunde. Das Limit wird automatisch so angepasst, dass es dem Limit 1/30 der aktiven Serie pro Workspace entspricht, bis zu 1.666.666.

Name	Standard	Anpas	Description
Unterdrückungsregeln in der Alert-Manager-Definitionsdatei	Jede unterstützte Region: 100	Yes (Ja)	Die maximale Anzahl von Unterdrückungsregeln in der Alert-Manager-Definitionsdatei.
Labelgröße	Jede unterstützte Region: 7	Nein	Die maximale Gesamtgröße aller für eine Serie akzeptierten Beschriftungen und Labelwerte in Kilobyte.
LabelSet Grenzwerte pro Arbeitsbereich	Jede unterstützte Region: 100	Yes (Ja)	Die maximale Anzahl von Labelset-Grenzwerten, die pro Workspace erstellt werden können.
Labels pro metrischer Serie	Jede unterstützte Region: 150	Ja	Anzahl der Etiketten pro metrischer Serie.
Maximale Anzahl an nativen Histogramm-Buckets	Jede unterstützte Region: 200	Nein	Die maximale Gesamtzahl der Buckets, die in einer Native-Histogramm-Stichprobe zulässig sind. Die Auflösung eines Histogramms mit mehr Buckets wird reduziert, bis die Anzahl der Buckets innerhalb des Grenzwerts liegt.

Name	Standard	Anpas	Description
Länge der Metadaten	Jede unterstützte Region: 1	Nein	Die maximal zulässige Länge für metrische Metadaten in Kilobyte. Metadaten beziehen sich auf den Namen, den Typ, die Einheit und den Hilfetext der Metrik.
Metadaten pro Metrik	Jede unterstützte Region: 10	Nein	Die Anzahl von Metadaten pro Metrik. Hinweis: Wenn der Grenzwert erreicht ist, wird eine Stichprobe von Metriken aufgezeic hnet, Metadaten, die den Grenzwert überschreiten, werden jedoch gelöscht.
Natives Histogramm, aktive Serien pro Workspace	Jede unterstützte Region: 5 000 000	Ja	Die Anzahl der eindeutig en aktiven Serien mit nativem Histogramm pro Workspace. Eine Serie ist aktiv, wenn in den letzten 2 Stunden ein Beispiel gemeldet wurde. Das Limit wird automatisch so angepasst, dass es 10 Prozent des Limits für aktive Workspace-Serien oder den Standardwert beträgt, je nachdem, welcher Wert höher ist.

Name	Standard	Anpas	Description
Aufnahmerate für native Histogramme pro Workspace	Jede unterstützte Region: 166.666	Ja	Aufnahmerate für native Histogramm-Proben pro Workspace pro Sekunde. Das Limit wird automatisch so angepasst, dass es 10 Prozent des Limits für die Aufnahmerate im Workspace beträgt, oder auf den Standardwert, je nachdem, welcher Wert höher ist.
Knoten im Alert-Manager-Routingbaum	Jede unterstützte Region: 100	Yes (Ja)	Die maximale Anzahl der Knoten im Alert-Manager-Routingbaum.
Anzahl der API-Operationen pro Region in Transaktionen pro Sekunde	Jede unterstützte Region: 10	Yes (Ja)	Die maximale Anzahl von API-Vorgängen pro Sekunde pro Region für alle Amazon Managed Service for Prometheus APIs, einschließlich Workspace-CRUD-APIs, Tagging-APIs, Regelgruppen-Namespace-CRUD-APIs und CRUD-APIs für Alert Manager-Definitionen.
Anzahl der GetLabels und GetMetricMetadata API-Operationen pro Workspace in Transaktionen pro Sekunde	Jede unterstützte Region: 10	Nein	Die maximale Anzahl von GetSeries GetLabels und GetMetricMetadata Prometheus-compatible API-Operationen pro Sekunde pro Workspace.

Name	Standard	Anpas	Description
Anzahl der QueryMetrics API-Operationen pro Workspace in Transaktionen pro Sekunde	Jede unterstützte Region: 300	Ja	Die maximale Anzahl von QueryMetrics Prometheus-compatible API-Vorgängen pro Sekunde pro Workspace.
Anzahl der RemoteWrite API-Operationen pro Workspace in Transaktionen pro Sekunde	Jede unterstützte Region: 3 000	Ja	Die maximale Anzahl von RemoteWrite Prometheus-compatible API-Vorgängen pro Sekunde pro Workspace.
Anzahl anderer Prometheus-compatible API-Operationen pro Workspace in Transaktionen pro Sekunde	Jede unterstützte Region: 100	Nein	Die maximale Anzahl von API-Vorgängen pro Sekunde pro Workspace für alle anderen Prometheus-compatible APIs ListAlerts ListRules, einschließlich usw.
Aufnahmerate pro Workspace, die nicht in der richtigen Reihenfolge sind	Jede unterstützte Region: 83.333	Ja	Aufnahmerate von Proben, die nicht in der richtigen Reihenfolge sind, pro Arbeitsbereich pro Sekunde. Sofern es nicht überschrieben wird, wird das Limit automatisch auf 5 Prozent der Aufnahmerate pro Workspace angepasst.

Name	Standard	Anpas	Description
Zeitfenster pro Arbeitsbereich nicht in der richtigen Reihenfolge	Jede unterstützte Region: 600	Ja	Das maximale Zeitfenster für Proben, die nicht in der richtigen Reihenfolge sind, pro Arbeitsbereich, in Sekunden. Das standardmäßige Zeitfenster, in dem ein neuer Arbeitsbereich erstellt wird, beträgt 60 Sekunden, und es kann bis zu einem Maximum von 600 Sekunden konfiguriert werden.
Bytes für sofortige Abfragen abfragen	Jede unterstützte Region: 5	Nein	Die maximale Anzahl an Byte, die mit einer einzigen Sofortabfrage gescannt werden können, in Gigabyte.
Byte für Bereichsabfragen abfragen	Jede unterstützte Region: 5	Nein	Die maximale Anzahl an Byte, die pro 24-Stunden-Intervall in einer einzelnen Bereichsabfrage gescannt werden können, in Gigabyte.
Abfragebeispiele	Jede unterstützte Region: 50 000 000	Nein	Die maximale Anzahl von Samples, die pro 24-Stunden-Intervall in einer einzelnen Bereichsabfrage oder einer einzelnen Sofortabfrage gescannt werden können.

Name	Standard	Anpas	Description
Abfrageserie abgerufen	Jede unterstützte Region: 12.000.000	Nein	Die maximale Anzahl von Serien, die pro 24-Stunden-Intervall in einer einzelnen Bereichsabfrage oder einer einzelnen Sofortabfrage gescannt werden können.
Zeitraum für die Abfrage in Tagen	Jede unterstützte Region: 95	Nein	Der maximale Zeitbereich von QueryMetrics GetSeries, und GetLabels APIs.
Anforderungsgröße	Jede unterstützte Region: 1	Nein	Die maximale Anforderungsgröße für die Aufnahme oder Abfrage in Megabyte.
Bewertungsintervall für die Regel	Jede unterstützte Region: 30	Yes (Ja)	Das minimale Regelauswertungsintervall einer Regelgruppe pro Workspace in Sekunden.
Größe der Definitionsdatei des Regelgruppen-Namespace	Jede unterstützte Region: 1 000 000	Nein	Die maximale Größe einer Regelgruppen-Namespace-Definitionsdatei in Byte.
Regeln pro Workspace	Jede unterstützte Region: 2.000	Ja	Die maximale Anzahl von Regeln pro Workspace.

Name	Standard	Anpas	Description
Stille Zeiten pro Arbeitsbereich	Jede unterstützte Region: 1 000	Ja	Maximale Anzahl von Stummschaltungen, einschließlich abgelaufe ner, aktiver und ausstehender Stille, pro Workspace.
Vorlagen in der Alert-Manager-Defi nitionsdatei	Jede unterstützte Region: 100	Yes (Ja)	Die maximale Anzahl der Vorlagen in der Alert-Man ager-Definitionsdatei.
Workspaces pro Region und Konto	Jede unterstützte Region: 25	Ja	Die maximale Anzahl von Workspaces pro Region.

Standardkontingente für aktive Serien

Die Arbeitsbereiche von Amazon Managed Service for Prometheus passen sich automatisch an Ihre Aufnahmenutzung an. Wenn Ihre Nutzung zunimmt, erhöht der Service automatisch Ihre Zeitreihenkapazität bis zum Standardkontingent.

Ihr Amazon Managed Service for Prometheus Workspace skaliert automatisch, basierend auf Ihrer Nutzung, und zwar auf zwei Arten:

1. Wenn Ihre durchschnittliche Nutzung von 30 Minuten unter 5 Millionen Serien liegt, verdoppelt sich die Kapazität (z. B. erhält ein Workspace mit 3,5 Mio. Nutzung eine Kapazität von 7 Mio.).
2. Wenn die Nutzung 5 Millionen Serien überschreitet, fügt der Workspace einen Puffer von 10 Millionen hinzu (ein Workspace mit 25 Millionen Auslastungen erhält beispielsweise eine Kapazität von 35 Millionen).

Amazon Managed Service for Prometheus weist automatisch mehr Kapazität zu, wenn Ihre Datenaufnahme steigt, bis Ihr Kontingent erreicht ist. Auf diese Weise wird sichergestellt, dass Ihr Workload nicht dauerhaft gedrosselt wird. Zu einer Drosselung kann es jedoch kommen, wenn Sie den Wert Ihres vorherigen Basiswerts, der in den letzten 30 Minuten berechnet wurde, um das Doppelte oder mehr als 10 Millionen überschritten haben. Um eine Drosselung zu vermeiden,

empfiehlt Amazon Managed Service for Prometheus, die Aufnahme schrittweise zu erhöhen, wenn sie über Ihren vorherigen Basiswert hinausgeht.

> **Note**
>
> Die Mindestkapazität für aktive Zeitreihen beträgt 2 Millionen, und es gibt keine Drosselung, wenn Sie weniger als 2 Millionen Zeitreihen haben.
> Um Ihr Standardkontingent zu überschreiten, können Sie eine Erhöhung des Kontingents beantragen.

Skalierung über das Standardkontingent hinaus

Wenn Sie eine Erhöhung des Kontingents über das Standardkontingent für aktive Serien hinaus beantragen, passt Amazon Managed Service for Prometheus Ihre Workspace-Kapazität entsprechend an. Wenn Sie die erhöhte Kapazität nicht vollständig nutzen, wird der Service den ungenutzten Teil im Laufe der Zeit zurückfordern. Wenn Ihre Nutzung zunimmt, wird der Workspace automatisch wieder skaliert.

Eine Drosselung kann jedoch auftreten, wenn Sie im Vergleich zu Ihrem vorherigen Basiswert, berechnet aus den letzten 2 Stunden, mehr als 50 Millionen aktive Zeitreihen mehr als verdoppeln oder überschreiten. Beispiel:

- Wenn Ihr Kontingent 100 Millionen beträgt und Ihr Basiswert 30 Millionen beträgt, können Sie ohne Drosselung innerhalb von 2 Stunden auf 60 Millionen hochskalieren.
- Wenn Ihr Kontingent 100 Millionen beträgt und Ihr Basiswert 50 Millionen beträgt, können Sie innerhalb von 2 Stunden ohne Drosselung auf die vollen 100 Millionen hochskalieren.

Drosselung der Aufnahme

Amazon Managed Service for Prometheus drosselt die Datenaufnahme für jeden Workspace, basierend auf Ihren aktuellen Limits. Dies trägt dazu bei, die Leistung des Workspace aufrechtzuerhalten. Wenn du das Limit überschreitest, siehst du das `DiscardedSamples` in den CloudWatch Kennzahlen (mit dem `rate_limited` Grund). Sie können CloudWatch damit Ihre Aufnahme überwachen und einen Alarm auslösen, der Sie warnt, wenn Sie kurz davor sind, die Drosselungsgrenzen zu erreichen. Weitere Informationen finden Sie unter Verwenden Sie

CloudWatch Metriken, um die Ressourcen von Amazon Managed Service for Prometheus zu überwachen.

Amazon Managed Service for Prometheus verwendet den Token-Bucket-Algorithmus, um die Aufnahmedrosselung zu implementieren. Mit diesem Algorithmus verfügt Ihr Konto über einen Bucket, der eine bestimmte Anzahl von Token enthält. Die Anzahl der Token im Bucket entspricht Ihrem Aufnahmelimit zu einer bestimmten Sekunde.

Jede aufgenommene Datenprobe entfernt ein Token aus dem Bucket. Wenn deine Bucket-Größe (Aufnahmerate pro Workspace) 1.000.000 beträgt, kann dein Workspace eine Million Datenproben in einer Sekunde aufnehmen. Wenn mehr als eine Million Samples aufgenommen werden müssen, wird der Vorgang gedrosselt und es werden keine weiteren Datensätze mehr aufgenommen. Zusätzliche Datenproben werden verworfen.

Der Eimer wird automatisch mit einer festgelegten Geschwindigkeit wieder aufgefüllt. Wenn der Bucket seine maximale Kapazität unterschreitet, wird ihm jede Sekunde eine festgelegte Anzahl von Tokens hinzugefügt, bis er seine maximale Kapazität erreicht hat. Wenn der Eimer voll ist, wenn die Nachfüllmarken eintreffen, werden sie weggeworfen. Der Eimer kann nicht mehr als die maximale Anzahl an Tokens aufnehmen. Die Nachfüllrate für die Probenaufnahme wird durch den Grenzwert für die Aufnahmerate pro Arbeitsplatz festgelegt. Wenn deine Aufnahmerate pro Workspace auf 170.000 festgelegt ist, beträgt die Nachfüllrate für den Bucket 170.000 Token pro Sekunde.

Wenn dein Workspace in einer Sekunde 1.000.000 Datenproben aufnimmt, wird dein Bucket sofort auf null Token reduziert. Der Bucket wird dann jede Sekunde mit 170.000 Token aufgefüllt, bis er seine maximale Kapazität von 1.000.000 Token erreicht hat. Wenn keine Aufnahme mehr erfolgt, wird der zuvor leere Eimer innerhalb von 6 Sekunden wieder auf seine maximale Kapazität zurückgesetzt.

> **Note**
>
> Die Aufnahme erfolgt in gebündelten Anfragen. Wenn Sie 100 Token zur Verfügung haben und eine Anfrage mit 101 Proben senden, wird die gesamte Anfrage abgelehnt. Amazon Managed Service for Prometheus akzeptiert Anfragen nicht teilweise. Wenn Sie einen Collector schreiben, können Sie Wiederholungsversuche verwalten (mit kleineren Batches oder nach Ablauf einer gewissen Zeit).

Sie müssen nicht warten, bis der Bucket voll ist, bevor Ihr Workspace weitere Datenproben aufnehmen kann. Du kannst Tokens verwenden, sobald sie dem Bucket hinzugefügt werden. Wenn Sie die Nachfüll-Token sofort verwenden, erreicht der Eimer nicht seine maximale Kapazität. Wenn

Sie beispielsweise den Bucket leer haben, können Sie weiterhin 170.000 Datenproben pro Sekunde aufnehmen. Der Bucket kann nur dann bis zur maximalen Kapazität aufgefüllt werden, wenn Sie weniger als 170.000 Datenproben pro Sekunde aufnehmen.

Zusätzliche Limits für erfasste Daten

Amazon Managed Service für Prometheus verfügt über zusätzliche Anfragen für erfasste Daten, die in den Workspace aufgenommen wurden. Diese sind nicht einstellbar.

- Metrische Beispiele, die älter als 1 Stunde sind, werden nicht erfasst.
- Jedes Beispiel und alle Metadaten müssen einen Metriknamen haben.
- Native Histogrammproben müssen einen Schemawert zwischen -4 und 8 haben. Stichproben mit einem Schema außerhalb dieses Bereichs werden verworfen.
- Native Histogrammproben dürfen eine Größe von 2048 Byte nicht überschreiten. Stichproben, die diesen Grenzwert überschreiten, werden verworfen.

API-Referenz für Amazon Managed Service für Prometheus

Amazon Managed Service für Prometheus bietet zwei Arten von: APIs

1. Amazon Managed Service for Prometheus APIs — APIs Damit können Sie Ihre Amazon Managed Service for Prometheus Workspaces erstellen und verwalten, einschließlich Operationen für Workspaces, Scraper, Alert Manager-Definitionen, Regelgruppen, Namespaces und Protokollierung. Sie verwenden die, die für verschiedene Programmiersprachen verfügbar sind AWS SDKs, um mit diesen zu interagieren. APIs
2. Prometheus kompatibel APIs — Amazon Managed Service for Prometheus unterstützt HTTP APIs , das mit Prometheus kompatibel ist. Diese APIs ermöglichen die Erstellung benutzerdefinierter Anwendungen, die Automatisierung von Workflows, die Integration mit anderen Diensten oder Tools sowie die Abfrage und Interaktion mit Ihren Überwachungsdaten mithilfe der Prometheus-Abfragesprache (PromQL).

In diesem Abschnitt werden die API-Vorgänge und Datenstrukturen aufgeführt, die von Amazon Managed Service für Prometheus unterstützt werden.

Informationen zu den Kontingenten für Serien, Labels und API-Anfragen finden Sie unter Amazon Managed Service for Prometheus Service-Kontingente im Amazon Managed Service for Prometheus User Guide.

Themen

- Amazon Managed Service für Prometheus APIs
- Prometheus-compatible APIs

Amazon Managed Service für Prometheus APIs

Amazon Managed Service for Prometheus bietet API-Operationen zur Erstellung und Verwaltung Ihrer Amazon Managed Service for Prometheus-Arbeitsbereiche. Dazu gehören APIs für Workspaces, Scraper, Alert Manager-Definitionen, Regelgruppen, Namespaces und Protokollierung.

Ausführliche Informationen zum Amazon Managed Service für Prometheus APIs finden Sie in der Amazon Managed Service for Prometheus API-Referenz.

Verwenden von Amazon Managed Service für Prometheus mit einem SDK AWS

AWS Software Development Kits (SDKs) sind für viele gängige Programmiersprachen verfügbar. Jedes SDK bietet eine API, Codebeispiele und Dokumentation, die es Entwicklern erleichtern, AWS Anwendungen in ihrer bevorzugten Sprache zu erstellen. Eine Liste von Tools SDKs und Tools nach Sprachen finden Sie AWS im AWS Developer Center unter Tools to Build On.

> **SDK-Versionen**
>
> Wir empfehlen Ihnen, die neueste Version des AWS SDK und alle anderen Versionen zu verwenden SDKs, die Sie in Ihren Projekten verwenden, und diese auf dem neuesten SDKs Stand zu halten. Das AWS SDK bietet Ihnen die neuesten Features und Funktionen sowie Sicherheitsupdates.

Prometheus-compatible APIs

Amazon Managed Service for Prometheus unterstützt die folgenden Prometheus-compatible APIs.

Weitere Informationen zur Verwendung von Prometheus-compatible APIs finden Sie unter. Abfragen mithilfe von APIs Prometheus-compatible

Themen

- CreateAlertManagerAlerts
- DeleteAlertManagerSilence
- GetAlertManagerStatus
- GetAlertManagerSilence
- GetLabels
- GetMetricMetadata
- GetSeries
- ListAlerts
- ListAlertManagerAlerts
- ListAlertManagerAlertGroups

- ListAlertManagerReceivers
- ListAlertManagerSilences
- ListRules
- PutAlertManagerSilences
- QueryMetrics
- RemoteWrite

CreateAlertManagerAlerts

Der Vorgang `CreateAlertManagerAlerts` erstellt einen Alarm im Workspace.

Gültige HTTP-Verben:

`POST`

Gültige URI:

`/workspaces/workspaceId/alertmanager/api/v2/alerts`

URL-Abfrageparameter:

`alerts` Ein Array von Objekten, wobei jedes Objekt für einen Alarm steht. Im Folgenden wird ein Beispiel für ein Alarm-Objekt gezeigt:

```
[
  {
    "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"
  }
]
```

Beispielanforderung

```
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/"
  }
]
```

Beispielantwort

```
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` löscht eine Alarmstummschaltung.

Gültige HTTP-Verben:

`DELETE`

Gültige URI:

`/workspaces/`*`workspaceId`*`/alertmanager/api/v2/silence/`*`silenceID`*

URL-Abfrageparameter: keine

Beispielanforderung

```
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
```

Beispielantwort

```
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` ruft Informationen über den Status des Alert Managers ab.

Gültige HTTP-Verben:

- `GET`

Gültige URI:

- `/workspaces/workspaceId/alertmanager/api/v2/status`

URL-Abfrageparameter: keine

Beispielanforderung

```
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
```

Beispielantwort

```
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` ruft Informationen über eine Alarmstummschaltung ab.

Gültige HTTP-Verben:

- `GET`

Gültige URI:

- `/workspaces/workspaceId/alertmanager/api/v2/silence/silenceID`

URL-Abfrageparameter: keine

Beispielanforderung

```
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
```

Beispielantwort

```
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

Der `GetLabels` Vorgang ruft die mit einer Zeitreihe verknüpften Labels ab.

Gültige HTTP-Verben:

`GET`, `POST`

Gültige URI:

`/workspaces/`*`workspaceId`*`/api/v1/labels`

`/workspaces/`*`workspaceId`*`/api/v1/label/`*`label-name`*`/values` Dieser URI unterstützt nur GET-Anfragen.

URL-Abfrageparameter:

`match[]=<series_selector>`Wiederholtes Serienauswahlargument, das die Serie auswählt, aus der die Labelnamen gelesen werden sollen. Optional.

`start=<rfc3339 | unix_timestamp>`Startzeitstempel. Optional.

`end=<rfc3339 | unix_timestamp>`Endzeitstempel. Optional.

Beispielanfrage für **`/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
```

Beispielantwort für **`/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",
        ...
    ]
}
```

Beispielanfrage für **/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
```

Beispielantwort für **/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

Der `GetMetricMetadata`-Vorgang ruft Metadaten zu Metriken ab, die derzeit von Zielen gescraped werden. Es werden keine Zielinformationen bereitgestellt.

Der Datenbereich des Abfrageergebnisses besteht aus einem Objekt, bei dem jeder Schlüssel ein Metrikname und jeder Wert eine Liste einmaliger Metadatenobjekte ist, die für diesen Metriknamen für alle Ziele verfügbar gemacht werden.

Gültige HTTP-Verben:

`GET`

Gültige URI:

`/workspaces/`*`workspaceId`*`/api/v1/metadata`

URL-Abfrageparameter:

`limit=<number>` Die maximale Anzahl der zurückzugebenden Zeilen.

`metric=<string>` Ein Metrikname, nach dem Metadaten gefiltert werden sollen. Wenn Sie dies leer lassen, werden alle Metrik-Metadaten abgerufen.

Beispielanforderung

```
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
```

Beispielantwort

```
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

Der `GetSeries` Vorgang ruft eine Liste von Zeitreihen ab, die einem bestimmten Labelsatz entsprechen.

Gültige HTTP-Verben:

`GET`, `POST`

Gültige URI:

`/workspaces/`*`workspaceId`*`/api/v1/series`

URL-Abfrageparameter:

`match[]=<series_selector>` Wiederholtes Serienauswahlargument, das die zurückzugebende Serie auswählt. Mindestens ein `match[]` Argument muss angegeben werden.

`start=<rfc3339 | unix_timestamp>`Startzeitstempel. Optional

`end=<rfc3339 | unix_timestamp>`Endzeitstempel. Optional

Beispielanforderung

```
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
```

Beispielantwort

```
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

Der Vorgang `ListAlerts` ruft derzeit aktive Alarme im Workspace ab.

Gültige HTTP-Verben:

`GET`

Gültige URI:

`/workspaces/workspaceId/api/v1/alerts`

Beispielanforderung

```
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
```

Beispielantwort

```
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` ruft Informationen über die Alarme ab, die derzeit im Alert Manager im Workspace ausgelöst werden.

Gültige HTTP-Verben:

`GET`

Gültige URI:

`/workspaces/workspaceId/alertmanager/api/v2/alerts`

URL-Abfrageparameter:

Boolescher Wert `active`. Falls wahr, enthält die zurückgegebene Liste aktive Alarme. Der Standardwert ist true. Optional

Boolescher Wert `silenced`. Wenn der Wert „true“ ist, enthält die zurückgegebene Liste Alarmstummschaltungen. Der Standardwert ist true. Optional

Boolescher Wert `inhibited`. Wenn der Wert „true“ ist, enthält die zurückgegebene Liste gesperrte Alarme. Der Standardwert ist true. Optional

`filter` Ein Array von Strings. Eine Liste von Matchern, nach denen Alarme gefiltert werden sollen. Optional

`receiver` String. Ein regulärer Ausdruck, der mit Empfängern übereinstimmt, nach denen Alarme gefiltert werden sollen. Optional

Beispielanforderung

```
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
```

Beispielantwort

```
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

Der Vorgang `ListAlertManagerAlertGroups` ruft eine Liste von Alarmgruppen ab, die im Alert Manager im Workspace konfiguriert wurden.

Gültige HTTP-Verben:

`GET`

Gültige URI:

`/workspaces/workspaceId/alertmanager/api/v2/alerts/groups`

URL-Abfrageparameter:

Boolescher Wert `active`. Falls wahr, enthält die zurückgegebene Liste aktive Alarme. Der Standardwert ist true. Optional

Boolescher Wert `silenced`. Wenn der Wert „true“ ist, enthält die zurückgegebene Liste Alarmstummschaltungen. Der Standardwert ist true. Optional

Boolescher Wert `inhibited`. Wenn der Wert „true“ ist, enthält die zurückgegebene Liste gesperrte Alarme. Der Standardwert ist true. Optional

Boolescher Wert `muted`. Wenn der Wert wahr ist, enthält die zurückgegebene Liste stummgeschaltete Warnmeldungen. Der Standardwert ist true. Optional

`filter` Ein Array von Strings. Eine Liste von Matchern, nach denen Alarme gefiltert werden sollen. Optional

`receiver` String. Ein regulärer Ausdruck, der mit Empfängern übereinstimmt, nach denen Alarme gefiltert werden sollen. Optional

Beispielanforderung

```
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
```

Beispielantwort

```
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

Der `ListAlertManagerReceivers` Vorgang ruft Informationen über die im Alert Manager konfigurierten Empfänger ab.

Gültige HTTP-Verben:

`GET`

Gültige URI:

`/workspaces/workspaceId/alertmanager/api/v2/receivers`

URL-Abfrageparameter: keine

Beispielanforderung

```
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
```

Beispielantwort

```
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

Der Vorgang `ListAlertManagerSilences` ruft Informationen über die im Workspace konfigurierten Alarmstummschaltungen ab.

Gültige HTTP-Verben:

`GET`

Gültige URI:

`/workspaces/workspaceId/alertmanager/api/v2/silences`

URL-Abfrageparameter:

`filter` Ein Array von Strings. Eine Liste von Matchern, nach denen Stille gefiltert werden kann. Optional

Beispielanforderung

```
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
```

Beispielantwort

```
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

Der `ListRules` Vorgang ruft Informationen zu den im Workspace konfigurierten Regeln ab.

Eine einzelne Antwort enthält höchstens 100.000 Regeln. Da Regelgruppen niemals auf mehrere Seiten aufgeteilt werden, führt das Kürzen immer zu einer Regelgruppengrenze. Eine Antwort kann

daher weniger als 100.000 Regeln enthalten, wenn die Aufnahme der nächsten Regelgruppe die Obergrenze überschreitet. Wenn eine Kürzung erfolgt, enthält die Antwort die `groupNextToken` Eigenschaft. Um die nächste Seite abzurufen, übergeben Sie diesen Wert als `group_next_token` Abfrageparameter, den Sie zusammen mit `group_limit` senden müssen.

Gültige HTTP-Verben:

`GET`

Gültige URI:

`/workspaces/workspaceId/api/v1/rules`

URL-Abfrageparameter:

`type=<alert|record>`Gibt nur die Warnungsregeln (zum Beispiel`type=alert`) oder die Aufzeichnungsregeln (zum Beispiel,`type=record`) zurück. Wenn der Parameter fehlt oder leer ist, wird keine Filterung durchgeführt. Optional

`rule_name[]=<string>`Filtert Regeln anhand der angegebenen Namen. Wenn mehrere Filter bereitgestellt werden, muss der Regelname mindestens einem der angegebenen Filter entsprechen. Optional

`rule_group[]=<string>`Filtert Regelgruppen anhand der angegebenen Namen. Wenn mehrere Filter bereitgestellt werden, muss der Name der Regelgruppe mindestens einem der angegebenen Filter entsprechen. Optional

`file[]=<string>`Filtert Regelgruppen-Namespaces anhand der angegebenen Namen. Wenn mehrere Filter bereitgestellt werden, muss der Name des Regelgruppen-Namespaces mindestens einem der angegebenen Filter entsprechen. Optional

`exclude_alerts=<true|false>`Filtert Regeln ohne aktive Benachrichtigungen, wenn diese Einstellung auf `true` gesetzt ist. Optional

`match[]=<series_selector>`Gibt nur Regeln zurück, für die Labels konfiguriert wurden, die den Label-Selektoren entsprechen. Wenn der Parameter wiederholt wird, werden Regeln zurückgegeben, die mit einer der Gruppen von Labelselektoren übereinstimmen. Beachten Sie, dass sich der Abgleich auf die Labels in der Definition der einzelnen Regeln bezieht, nicht auf die Werte nach der Erweiterung der Vorlage (bei Warnungsregeln). Optional

`group_limit=<number>`Maximale Anzahl von Regelgruppen, die in einer einzigen Antwort zurückgegeben wurden. Verwenden Sie bei nachfolgenden Anfragen zur Paginierung die

`groupNextToken` Eigenschaft aus dieser Antwort, um den `group_next_token` Parameter festzulegen. Optional

`group_next_token=<string>`Das Paginierungstoken, das von einer vorherigen Anfrage zurückgegeben wurde, wenn Sie die `group_limit` Eigenschaft festgelegt haben oder wenn die Antwort die Obergrenze von 100.000 Regeln erreicht. Verwenden Sie dieses Token, um die verbleibenden Regelgruppen iterativ zu paginieren. Wenn Sie diesen Parameter verwenden, müssen Sie den `group_limit` Parameter auch senden. Optional

Beispielanforderung

```
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
```

Beispielantwort

```
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

Der Vorgang `PutAlertManagerSilences` erzeugt eine neue Alarmstummschaltung oder aktualisiert eine bestehende.

Gültige HTTP-Verben:

`POST`

Gültige URI:

`/workspaces/`*`workspaceId`*`/alertmanager/api/v2/silences`

URL-Abfrageparameter:

`silence`Ein Objekt, das die Stummschaltung darstellt. Das Format sieht wie folgt aus:

```
{
  "id": "string",
  "matchers": [
    {
      "name": "string",
      "value": "string",
      "isRegex": Boolean,
      "isEqual": Boolean
    }
  ],
  "startsAt": "timestamp",
```

```
  "endsAt": "timestamp",
  "createdBy": "string",
  "comment": "string"
}
```

Beispielanforderung

```
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"
}
```

Beispielantwort

```
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

Der `QueryMetrics` Vorgang wertet eine Sofortabfrage zu einem bestimmten Zeitpunkt oder über einen bestimmten Zeitraum aus.

Gültige HTTP-Verben:

`GET`, `POST`

Gültige URI:

`/workspaces/`*`workspaceId`*`/api/v1/query`Dieser URI wertet eine Sofortabfrage zu einem einzigen Zeitpunkt aus.

`/workspaces/`*`workspaceId`*`/api/v1/query_range`Dieser URI wertet eine Sofortabfrage über einen bestimmten Zeitraum aus.

URL-Abfrageparameter:

`query=<string>`Ein Abfrage-String mit einem Prometheus-Ausdruck. Wird sowohl in `query` als auch in `query_range` verwendet.

`time=<rfc3339 | unix_timestamp>` (Optional) Zeitstempel für die Auswertung, wenn Sie den `query` für eine Sofortabfrage zu einem einzigen Zeitpunkt verwenden.

`timeout=<duration>` (Optional) Timeout für die Evaluierung. Der Standardwert ist auf den Wert der `-query.timeout` Flag eingestellt und wird durch diesen begrenzt. Wird sowohl in `query` als auch in `query_range` verwendet.

`start=<rfc3339 | unix_timestamp>` Startzeitstempel, wenn Sie `query_range` einen bestimmten Zeitraum abfragen möchten.

`end=<rfc3339 | unix_timestamp>` Endzeitstempel, wenn Sie für `query_range` die Abfrage nach einem bestimmten Zeitraum verwenden.

`step=<duration | float>` Schrittweite der Auflösung im `duration` Format oder als `float` Anzahl von Sekunden abfragen. Verwenden Sie diese Option nur, wenn Sie `query_range` für Abfragen für einen bestimmten Zeitraum verwenden und dies für solche Abfragen erforderlich sind.

`max_samples_processed_warning_threshold=<integer>`(Optional) Legt den Warnschwellenwert für QSP (Query Samples Processed) fest. Wenn Abfragen diesen Schwellenwert erreichen, wird in der API-Antwort eine Warnmeldung zurückgegeben.

`max_samples_processed_error_threshold=<integer>>`(Optional) Legt den Fehlerschwellenwert für QSP (Query Samples Processed) fest. Abfragen, die diesen Schwellenwert überschreiten, werden mit einem Fehler zurückgewiesen und nicht berechnet. Wird verwendet, um übermäßige Abfragekosten zu vermeiden.

Duration (Dauer)

A `duration` in einer Prometheus-compatible API ist eine Zahl, unmittelbar gefolgt von einer der folgenden Einheiten:

- `ms` Millisekunden
- `s` Sekunden
- `m` Minuten
- `h` Stunden
- `d` Tage, vorausgesetzt, ein Tag hat immer 24 Stunden
- `w` Wochen, vorausgesetzt, eine Woche hat immer 7 Tage
- `y` Jahre, vorausgesetzt, ein Jahr hat immer 365 Tage

Beispielanforderung

```
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
```

Beispielantwort

```
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

Der `RemoteWrite`-Vorgang schreibt Metriken von einem Prometheus-Server in einem standardisierten Format in eine Remote-URL. In der Regel verwenden Sie einen vorhandenen Client, z. B. einen Prometheus-Server, um diesen Vorgang aufzurufen.

Gültige HTTP-Verben:

- `POST`

Gültige URI:

- `/workspaces/workspaceId/api/v1/remote_write`

URL-Abfrageparameter:

- Keine

`RemoteWrite` hat eine Erfassungsrate von 70.000 Proben pro Sekunde und eine Erfassungs-Burst-Größe von 1.000.000 Proben.

Beispielanforderung

```
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**
>
> Die Syntax des Anforderungstexts finden Sie in der Protokollpuffer-Definition unter https://github.com/prometheus/prometheus/blob/1c624c58ca934f618be737b4995e22051f5724c1/prompb/remote.pb.go#L64.

Beispielantwort

```
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
```

Benutzerhandbuch von Amazon Managed Service für Prometheus

In der folgenden Tabelle werden wichtige Aktualisierungen der Dokumentation im Benutzerhandbuch von Amazon Managed Service für Prometheus beschrieben. Um Benachrichtigungen über Aktualisierungen dieser Dokumentation zu erhalten, können Sie einen RSS-Feed abonnieren.

Änderung	Beschreibung	Datum
Die AmazonPrometheusScraperServiceRolePolicy verwaltete IAM-Richtlinie wurde aktualisiert	AWS verwaltete Sammler können jetzt als neuen Zieltyp gesammelte CloudWatch Metriken in Amazon Metrics-Datensätze schreiben.	14. Juli 2026
Unterstützung für native Histogramme hinzugefügt	Amazon Managed Service for Prometheus unterstützt jetzt die Aufnahme, Speicherung und Abfrage von systemeigenen Prometheus-Histogrammen. Es stehen neue CloudWatch Nutzungsmetriken zur Verfügung, mit denen die Nutzung nativer `NativeHistogramIngestionRate` Histogramme anhand von `NativeHistogramActiveSeries` Kontingentbeschränkungen überwacht werden kann. Eine neue Servicemetrik bietet Einblick in den Bucketverbrauch. `NativeHistogramIngestedBucketsRate` Weitere Informationen finden Sie unter Mit	10. Juni 2026

	CloudWatch Metriken überwachen.	
Unterstützung für Out-of-Order-Ingestion und Offset für Regelabfragen wurde hinzugefügt	Amazon Managed Service for Prometheus unterstützt jetzt die Konfiguration eines Zeitfensters außerhalb der Reihenfolge für die Annahme von Proben, die nicht in der chronologischen Reihenfolge ankommen, und einen Regelabfrageversatz, um eine globale Verzögerung vor Regelauswertungen einzuführen. Zwei neue CloudWatch Messwerte, `OutOfOrderIngestionRate` und, sind ebenfalls verfügbar, um die `OutOfOrderSampleAge` Aufnahme außerhalb der angegebenen Reihenfolge zu überwachen. Weitere Informationen findest du unter Konfiguriere deinen Workspace.	14. Mai 2026

Die Unterstützung für wurde gestartet PagerDuty	Amazon Managed Service for Prometheus bietet zusätzliche PagerDuty Integrationsunterstützung, die automatisierte Workflows zur Reaktion auf Vorfälle ermöglicht und sicherstellt, dass kritische Benachrichtigungen die richtigen Teammitglieder zur richtigen Zeit erreichen. Weitere Informationen finden Sie unter PagerDuty Als Alarmempfänger verwenden.	29. August 2025
Unterstützung für ressourcenbasierte Richtlinien hinzugefügt	Die folgenden API-Aktionen sind jetzt verfügbar: • DeleteResourcePolicy • DescribeResourcePolicy • PutResourcePolicy	15. August 2025
Aktualisierung der AmazonPrometheusConsoleFullAccess verwalteten IAM-Richtlinie.	Die AmazonPrometheusConsoleFullAccessRichtlinie wurde aktualisiert. Die `aps:CreateQueryLoggingConfiguration` ,`aps:UpdateQueryLoggingConfiguration` ,`aps:DeleteQueryLoggingConfiguration` , `aps:DescribeQueryLoggingConfiguration` Berechtigungen wurden der Richtlinie hinzugefügt.	5. Mai 2025

Die Bearbeitung von Regeldefinitionsdateien und Alert Manager-Konfigurationsdateien wurde in der Konsole hinzugefügt	Amazon Managed Service for Prometheus bietet Unterstützung für die Bearbeitung von Alert Manager-Konfigurationsdateien und Regeldefinitionsdateien von der Amazon Managed Service for Prometheus-Konsole aus.	16. Mai 2024
Einfacheres AWS Managed-Collector-Setup mit Zugriffseinträgen für Amazon EKS hinzugefügt	Amazon Managed Service for Prometheus bietet Unterstützung für Amazon EKS-Zugriffseinträge, um die Einrichtung von AWS Managed Collectors zu vereinfachen. Die AmazonPrometheusScraperServiceRolePolicyverwaltete Richtlinie für AWS verwaltete Collectors wurde aktualisiert und ermöglicht nun das Löschen von Zugangseinträgen, die nicht mehr verwendet werden.	2. Mai 2024
Verschieben Sie die AWS API in ein separates API-Referenzhandbuch	Die Amazon Managed Service for Prometheus AWS APIs sind jetzt in einer eigenen Referenz, der Amazon Managed Service for Prometheus API-Referenz, verfügbar. Prometheus-compatible APIs werden weiterhin im Amazon Managed Service for Prometheus User Guide dokumentiert.	7. Februar 2024

Vom Kunden verwaltete Schlüssel für Workspace-Verschlüsselung hinzugefügt	Amazon Managed Service for Prometheus bietet Unterstützung für kundenverwaltete Schlüssel für die Workspace-Verschlüsselung. Weitere Informationen finden Sie unter Verschlüsselung im Ruhezustand.	21. Dezember 2023
Neue Berechtigungen wurden hinzugefügt zu AmazonPrometheusFullAccess	Der AmazonPrometheusFullAccessverwalteten Richtlinie wurden neue Berechtigungen hinzugefügt, um die Erstellung AWS verwalteter Collectors für Amazon EKS-Cluster zu unterstützen.	26. November 2023
Neue verwaltete Richtlinie hinzugefügt, AmazonPrometheusScraperServiceLinkedRolePolicy	Es wurde eine neue verwaltete Richtlinie hinzugefügt, mit der AmazonPrometheusScraperServiceLinkedRolePolicy AWS verwaltete Sammler Metriken aus Amazon EKS-Clustern sammeln können.	26. November 2023
AWS Verwaltete Collectors wurden als Erfassungsmethode hinzugefügt	Amazon Managed Service für Prometheus bietet Unterstützung für AWS verwaltete Kollektoren.	26. November 2023
Unterstützung für die Integration mit Amazon Managed Grafana hinzugefügt	Amazon Managed Service für Prometheus bietet Unterstützung für die Integration mit Alarmen in Amazon Managed Grafana.	23. November 2022

Neue Berechtigungen wurden hinzugefügt zu AmazonPrometheusConsoleFullAccess	Der AmazonPrometheusConsoleFullAccessverwaltete Richtlinie wurden neue Berechtigungen hinzugefügt, um die Protokollierung von Alert Manager- und Lineal-Ereignissen in CloudWatch Logs zu unterstützen.	24. Oktober 2022
Die Amazon-EKS-Beobachtbarkeitslösung wurde hinzugefügt.	Amazon Managed Service for Prometheus fügt mithilfe von AWS Observability Accelerator eine neue Lösung hinzu. Weitere Informationen finden Sie unter Einsatz von AWS Obersability Accelerator.	14. Oktober 2022
Unterstützung für die Integration in die Kostenüberwachung von Amazon EKS wurde hinzugefügt.	Amazon Managed Service für Prometheus bietet Unterstützung für die Integration in die Amazon-EKS-Kostenüberwachung. Weitere Informationen finden Sie unter Integration in die Kostenüberwachung von Amazon EKS.	22. September 2022
Die Unterstützung für Alert Manager- und Ruler-Protokolle in Amazon CloudWatch Logs wurde eingeführt.	Amazon Managed Service for Prometheus startet Unterstützung für Alert Manager- und Ruler-Fehlerprotokolle in Amazon CloudWatch Logs. Weitere Informationen finden Sie unter Amazon CloudWatch Logs.	01. September 2022

Unterstützung für benutzerdefinierte Speicheraufbewahrung hinzugefügt.	Amazon Managed Service für Prometheus bietet individuelle Unterstützung für die Aufbewahrung von Speicherplatz pro Workspace, indem das Kontingent für diesen Workspace geändert wird. Weitere Informationen zu Kontingenten in Amazon Managed Service für Prometheus finden Sie unter Servicekontingente.	12. August 2022
Nutzungsmetriken zu Amazon CloudWatch hinzugefügt.	Amazon Managed Service for Prometheus bietet Unterstützung für das Senden von Nutzungsmetriken an Amazon CloudWatch. Weitere Informationen finden Sie unter CloudWatchAmazon-Metriken.	6. Mai 2022
Unterstützung für die Region Europa (London) hinzugefügt.	Amazon Managed Service für Prometheus unterstützt jetzt die Region Europa (London).	4. Mai 2022
Amazon Managed Service für Prometheus ist allgemein verfügbar und bietet Unterstützung für Regeln und Alert Manager.	Amazon Managed Service für Prometheus ist allgemein verfügbar. Er unterstützt auch den Regeln- und Alert-Manager. Weitere Informationen finden Sie unter Aufzeichnungsregeln und Alarmregeln und Vorlagensystem.	29. September 2021

Tag-Unterstützung hinzugefügt.	Amazon Managed Service für Prometheus unterstützt das Taggen von Workspaces von Amazon Managed Service für Prometheus.	7. September 2021
Die Quoten für aktive Serien und Erfassungsraten wurden erhöht.	Die Quote für aktive Serien wurde auf 1.000.000 und die Quote für die Erfassungsrate auf 70.000 Proben pro Sekunde erhöht.	22. Februar 2021
Vorversion für Amazon Managed Service für Prometheus.	Die Vorversion von Amazon Managed Service für Prometheus wurde veröffentlicht.	15. Dezember 2020

Die vorliegende Übersetzung wurde maschinell erstellt. Im Falle eines Konflikts oder eines Widerspruchs zwischen dieser übersetzten Fassung und der englischen Fassung (einschließlich infolge von Verzögerungen bei der Übersetzung) ist die englische Fassung maßgeblich.