aws

Benutzerhandbuch

Amazon Managed Service für Prometheus

Amazon Managed Service für Prometheus: Benutzerhandbuch

Table of Contents

Was Amazon Managed Service für Prometheus?

Amazon Managed Service für Prometheus ist ein serverloser, Prometheus-kompatibler Service zur Überwachung von Containermetriken, der es einfacher macht, Containerumgebungen im großen Maßstab sicher zu überwachen. 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 in AWS Sicherheitsdienste integrieren, um einen schnellen und sicheren Zugriff auf Daten zu ermöglichen.

Amazon Managed Service für Prometheus ist so konzipiert, dass es mithilfe von Bereitstellungen in mehreren Multi-AZ-Bereitstellungen hochverfügbar ist. 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. Bei dieser Länge handelt es sich um ein einstellbares Kontingent.

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 aps-workspaces.us-east-2.amazonaws.com	HTTPS HTTPS	
USA Ost (Nord-Virginia)	us-east-1	aps.us-east-1.amazonaws.com aps-workspaces.us-east-1.amazonaws.com	HTTPS HTTPS	
USA West (Oregon)	us-west-2	aps.us-west-2.amazonaws.com aps-workspaces.us-west-2.amazonaws.com	HTTPS HTTPS	
Asien-Pazifik (Mumbai)	ap-south-1	aps.ap-south-1.amazonaws.com aps-workspaces.ap-south-1.amazonaws.com	HTTPS HTTPS	
Asien-Pazifik (Seoul)	ap-northeast-2	aps.ap-northeast-2.amazonaws.com aps-workspaces.ap-northeast-2.amazonaws.com	HTTPS HTTPS	
Asien-Pazifik (Singapur)	ap-southeast-1	aps.ap-southeast-1.amazonaws.com aps-workspaces.ap-southeast-1.amazonaws.com	HTTPS HTTPS	
Asien-Pazifik (Sydney)	ap-southeast-2	aps.ap-southeast-2.amazonaws.com aps-workspaces.ap-southeast-2.amazonaws.com	HTTPS HTTPS	
Asien-Pazifik (Tokio)	ap-northeast-1	aps.ap-northeast-1.amazonaws.com aps-workspaces.ap-northeast-1.amazonaws.com	HTTPS HTTPS	

Name der Region	Region	Endpunkt	Protocol (Protokoll)
Europa (Frankfurt)	eu-central-1	aps.eu-central-1.amazonaws.com aps-workspaces.eu-central-1.amazonaws.com	HTTPS HTTPS
Europa (Irland)	eu-west-1	aps.eu-west-1.amazonaws.com aps-workspaces.eu-west-1.amazonaws.com	HTTPS HTTPS
Europa (London)	eu-west-2	aps.eu-west-2.amazonaws.com aps-workspaces.eu-west-2.amazonaws.com	HTTPS HTTPS
Europa (Paris)	eu-west-3	aps.eu-west-3.amazonaws.com aps-workspaces.eu-west-3.amazonaws.com	HTTPS HTTPS
Europa (Stockholm)	eu-north-1	aps.eu-north-1.amazonaws.com aps-workspaces.eu-north-1.amazonaws.com	HTTPS HTTPS
Südamerika (São Paulo)	sa-east-1	aps.sa-east-1.amazonaws.com aps-workspaces.sa-east-1.amazonaws.com	HTTPS HTTPS

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 den Cost Explorer und die AWS Kosten- und Nutzungsberichte verwenden, um Ihre Gebühren zu überwachen. Weitere Informationen finden Sie unter Untersuchen Ihrer Daten mit dem Cost Explorer und Was sind AWS Kosten- und Nutzungsberichte.

Premium-Support

Wenn Sie eine Stufe der AWS Premium-Supportpläne abonnieren, gilt Ihr Premium-Support für Amazon Managed Service für Prometheus.

Erste Schritte

In diesem Abschnitt wird erklärt, wie Sie schnell Workspaces für Amazon Managed Service für Prometheus erstellen, die Erfassung von Prometheus-Metriken in diese Workspaces einrichten und diese Metriken abfragen können.

Es enthält auch Informationen zur Einrichtung eines AWS-Konto, falls Sie noch nicht damit vertraut sind AWS.

Themen

Einrichtung

Erledigen Sie die Aufgaben in diesem Abschnitt, um sich zum ersten Mal mit der Einrichtung vertraut AWS zu machen. Wenn Sie bereits ein AWS Konto haben, fahren Sie mit fortEinen Workspace erstellen.

Wenn Sie sich für registrieren AWS, hat Ihr AWS Konto automatisch Zugriff auf alle Services AWS, einschließlich Amazon Managed Service für Prometheus. Es werden jedoch nur die Services in Rechnung gestellt, die Sie tatsächlich nutzen.

Themen

Melden Sie sich an für ein AWS-Konto

Wenn Sie noch keine haben AWS-Konto, führen Sie die folgenden Schritte aus, um eine zu erstellen.

Um sich für eine anzumelden AWS-Konto

1. Öffnen Sie https://portal.aws.amazon.com/billing/signup.

2. Folgen Sie den Online-Anweisungen.

   Bei der Anmeldung müssen Sie auch einen Telefonanruf entgegennehmen und einen Verifizierungscode über die Telefontasten eingeben.

   Wenn Sie sich für eine anmelden AWS-Konto, Root-Benutzer des AWS-Kontoswird eine erstellt. Der Root-Benutzer hat Zugriff auf alle AWS-Services und Ressourcen des Kontos. Aus Sicherheitsgründen sollten Sie einem Benutzer Administratorzugriff zuweisen und nur den Root-Benutzer verwenden, um Aufgaben auszuführen, für die Root-Benutzerzugriff erforderlich ist.

AWS sendet Ihnen nach Abschluss des Anmeldevorgangs eine Bestätigungs-E-Mail. Sie können jederzeit Ihre aktuelle Kontoaktivität anzeigen und Ihr Konto verwalten. Rufen Sie dazu https://aws.amazon.com/ auf und klicken Sie auf Mein Konto.

Erstellen Sie einen Benutzer mit Administratorzugriff

Nachdem Sie sich für einen angemeldet haben AWS-Konto, sichern Sie Ihren Root-Benutzer des AWS-Kontos AWS IAM Identity Center, aktivieren und erstellen Sie einen Administratorbenutzer, sodass Sie den Root-Benutzer nicht für alltägliche Aufgaben verwenden.

Sichern Sie Ihre Root-Benutzer des AWS-Kontos

1. Melden Sie sich AWS Management Consoleals Kontoinhaber an, indem Sie Root-Benutzer auswählen und Ihre AWS-Konto E-Mail-Adresse eingeben. Geben Sie auf der nächsten Seite Ihr Passwort ein.

   Hilfe bei der Anmeldung mit dem Root-Benutzer finden Sie unter Anmelden als Root-Benutzer im AWS-Anmeldung Benutzerhandbuch zu.
2. Aktivieren Sie die Multi-Faktor-Authentifizierung (MFA) für den Root-Benutzer.

   Anweisungen finden Sie unter Aktivieren eines virtuellen MFA-Geräts für Ihren AWS-Konto Root-Benutzer (Konsole) im IAM-Benutzerhandbuch.

Erstellen Sie einen Benutzer mit Administratorzugriff

1. Aktivieren Sie das IAM Identity Center.

   Anweisungen finden Sie unter Aktivieren AWS IAM Identity Center im AWS IAM Identity Center Benutzerhandbuch.

2. Gewähren Sie einem Benutzer in IAM Identity Center Administratorzugriff.

   Ein Tutorial zur Verwendung von IAM-Identity-Center-Verzeichnis als Identitätsquelle finden Sie unter Benutzerzugriff mit der Standardeinstellung konfigurieren IAM-Identity-Center-Verzeichnis im AWS IAM Identity Center Benutzerhandbuch.

Melden Sie sich als Benutzer mit Administratorzugriff an

- Um sich mit Ihrem IAM-Identity-Center-Benutzer anzumelden, verwenden Sie die Anmelde-URL, die an Ihre E-Mail-Adresse gesendet wurde, als Sie den IAM-Identity-Center-Benutzer erstellt haben.

  Hilfe bei der Anmeldung mit einem IAM Identity Center-Benutzer finden Sie im AWS-Anmeldung Benutzerhandbuch unter Anmeldung beim AWS Zugriffsportal.

Weisen Sie weiteren Benutzern Zugriff zu

1. Erstellen Sie in IAM Identity Center einen Berechtigungssatz, der der bewährten Methode zur Anwendung von Berechtigungen mit den geringsten Rechten folgt.

   Anweisungen finden Sie im Benutzerhandbuch unter Einen Berechtigungssatz erstellen.AWS IAM Identity Center
2. Weisen Sie Benutzer einer Gruppe zu und weisen Sie der Gruppe dann Single Sign-On-Zugriff zu.

   Anweisungen finden Sie im AWS IAM Identity Center Benutzerhandbuch unter Gruppen hinzufügen.

Einen Workspace erstellen

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**
>
> Detaillierte Informationen zum Erstellen eines Modells finden Sie in Einen Workspace erstellen.

So erstellen Sie einen Workspace in Amazon Managed Service für Prometheus

1. Öffnen Sie die Konsole von Amazon Managed Service für Prometheus 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 Workspaces könnten denselben Alias haben, aber alle Workspaces haben einmalige Workspace-IDs, die von Amazon Managed Service für Prometheus generiert werden.
3. (Optional) Um Tags zum Namespace hinzuzufügen, 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-Write-Vorgänge 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 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.

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.

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 Erfassen Sie Metriken in Ihren Workspace.

> **Note**
>
> In einen Workspace aufgenommene Metriken werden standardmäßig 150 Tage lang gespeichert und dann automatisch gelöscht. Bei dieser Länge handelt es sich um ein einstellbares Kontingent.

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 als Agent Metriken an Amazon Managed Service für Prometheus zu senden. Für diese Methode müssen die folgenden Voraussetzungen erfüllt sein:

- Sie benötigen einen Amazon-EKS-Cluster, von dem der neue Prometheus-Server Metriken sammelt.
- 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. Ersetzen Sie *prometheus-agent-namespace* durch den Namen, den Sie für diesen Namespace wünschen.

```
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 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 Richten Sie Servicerollen für die Erfassung von Metriken aus Amazon EKS-Clustern ein, 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.

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

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

```
        capacity: 2500
```

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

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

```
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 Metriken von Amazon Managed Service für Prometheus abzufragen oder die APIs von Amazon Managed Service für Prometheus zu verwenden, finden Sie unter Abfragen Ihrer Prometheus-Metriken.

Sie führen Ihre Abfragen mit der standardmäßigen 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 Integration von Alerts mit 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.

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

Einen Workspace erstellen

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 Konsole von Amazon Managed Service für Prometheus unter https://console.aws.amazon.com/prometheus/.
2. Wählen Sie Erstellen.
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 zum Verschlüsseln der in Ihrem Workspace gespeicherten Daten zu verwenden, können Sie die Option Verschlüsselungseinstellungen anpassen auswählen und den zu verwendenden AWS KMS 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.

Einen Workspace bearbeiten

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 Konsole von Amazon Managed Service für Prometheus 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 Ihren Workspace-ARN

Sie finden den ARN Ihres Workspace in Amazon Managed Service für Prometheus entweder über die Konsole oder AWS CLI.

Finden Sie Ihren Workspace-ARN mithilfe der Konsole von Amazon Managed Service für Prometheus wie folgt

1. Öffnen Sie die Konsole von Amazon Managed Service für Prometheus 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.

   Der Workspace-ARN wird unter ARN angezeigt.

Geben Sie den folgenden Befehl ein, AWS CLI um den ARN für die Suche nach Ihrem Workspace zu verwenden.

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

Suchen Sie den Wert `arn` in den Ergebnissen.

Einen Workspace löschen

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 Konsole von Amazon Managed Service für Prometheus 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.

Erfassen Sie Metriken in Ihren Workspace

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. Diese Länge wird durch ein einstellbares Kontingent gesteuert.

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

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, damit Prometheus-kompatible Scraper darauf zugreifen können.

> **Note**
>
> Viele Technologien und Anwendungen, die in Kubernetes-Umgebungen ausgeführt werden, bieten Prometheus-kompatible 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, agentenlosen Scraper oder Kollektor, der automatisch Prometheus-kompatible Metriken erkennt und abruft. 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 Collectoren arbeiten 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

- Verwenden eines verwalteten Collectors AWS
- Was sind Prometheus-kompatible Metriken?

Verwenden eines verwalteten Collectors AWS

Um einen Kollektor von Amazon Managed Service für Prometheus zu verwenden, müssen Sie einen Scraper erstellen, der Metriken in Ihrem Amazon-EKS-Cluster erkennt und abruft.

- 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 der AWS API oder mithilfe der erstellen. AWS CLI

> **Note**
>
> Amazon Managed Service for Prometheus Workspaces, die mit vom Kunden verwalteten Schlüsseln erstellt wurden, können keine verwalteten Collectors für die AWS Erfassung verwenden.

Ein Kollektor von Amazon Managed Service für Prometheus erfasst Metriken, die mit Prometheus kompatibel sind. Weitere Informationen zu Prometheus-kompatiblen Metriken finden Sie unter Was sind Prometheus-kompatible Metriken?.

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

Themen

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.

Ein Scraper wird automatisch für Sie erstellt, wenn Sie über die Amazon-EKS-Konsole einen Amazon-EKS-Cluster erstellen. In einigen Situationen möchten Sie jedoch möglicherweise selbst einen Scraper erstellen. Zum Beispiel, wenn Sie einem vorhandenen Amazon EKS-Cluster einen AWS verwalteten Collector hinzufügen möchten oder wenn Sie die Konfiguration eines vorhandenen Collectors ändern möchten.

Sie können einen Scraper entweder mithilfe der AWS API oder der AWS CLI erstellen.

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.

> **Note**
>
> Der Cluster wird dem Scraper über seinen Amazon-Ressourcennamen (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. Sie löschen Scraper getrennt vom Löschen des Clusters.

AWS API

Um einen Scraper mithilfe der API zu erstellen AWS

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

> **Note**
>
> 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 von finden Sie `scrapeConfiguration` unterScraper-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

Um einen Scraper mit dem zu erstellen AWS CLI

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

> **Note**
>
> Sie müssen mindestens zwei Subnetze in mindestens zwei Availability Zones einfügen.

Das `scrape-configuration` ist eine Prometheus-Konfigurations-YAML-Datei, die Base64-codiert ist. Mit dem `get-default-scraper-configuration` Befehl können Sie eine Allzweckkonfiguration herunterladen. Weitere Informationen zum Format von finden Sie `scrape-configuration` unterScraper-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 CreateScraperAPI-Operation.
- Listen Sie Ihre vorhandenen Scraper mit der ListScrapersAPI-Operation auf.
- Löschen Sie einen Scraper mit der DeleteScraperAPI-Operation.
- Erfahren Sie mehr über einen Scraper mit der DescribeScraperAPI-Operation.
- Holen Sie sich eine Allzweckkonfiguration für Scraper mit dem GetDefaultScraperConfigurationAPI-Betrieb.

> **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.

Häufige Fehler beim Erstellen von Scrapern

Im Folgenden sind die häufigsten Probleme beim Versuch, einen neuen Scraper zu erstellen, aufgeführt.

- Erforderliche AWS Ressourcen sind nicht vorhanden. Die angegebene Sicherheitsgruppe, das Subnetz und der Amazon EKS-Cluster müssen vorhanden sein.
- Ungenügender IP-Adressraum. In jedem Subnetz, das Sie an die `CreateScraper` API übergeben, muss mindestens eine IP-Adresse verfügbar sein.

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-Zugriffseinträge, um Sammlern von Amazon Managed Service for Prometheus automatisch Zugriff auf Ihren Cluster zu gewähren.
- Konfigurieren Sie Ihren Amazon EKS-Cluster manuell für verwaltetes Metrik-Scraping.

In den folgenden Themen werden die einzelnen Themen ausführlicher beschrieben.

Amazon EKS für den Scraper-Zugriff mit Zugriffseinträgen konfigurieren

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 entweder `API` auf oder eingestellt sein. `API_AND_CONFIG_MAP` Dies ist in der Amazon EKS-Konsole auf der Registerkarte Zugriffskonfiguration der Cluster-Details sichtbar. Weitere Informationen finden Sie unter Erlauben

des Zugriffs von IAM-Rollen oder Benutzern auf das Kubernetes-Objekt in Ihrem Amazon EKS-Cluster im Amazon EKS-Benutzerhandbuch.

Sie können den Scraper bei der Erstellung des Clusters oder nach der Erstellung des Clusters erstellen:

- Beim Erstellen eines Clusters — Sie können diesen Zugriff konfigurieren, wenn Sie einen Amazon EKS-Cluster über die Amazon EKS-Konsole erstellen (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 nach der Erstellung eines Clusters — Wenn Ihr Amazon EKS-Cluster bereits existiert, setzen Sie den Authentifizierungsmodus entweder auf `API` oder. Für alle Scraper`API_AND_CONFIG_MAP`, die Sie über die Amazon Managed Service for Prometheus API oder CLI erstellen, wird automatisch die richtige Zugriffsrichtlinie für Sie erstellt, und die Scraper haben Zugriff auf Ihren Cluster.

Die Zugangsrichtlinie wurde erstellt

Wenn Sie einen Scraper erstellen und Amazon Managed Service for Prometheus eine Zugriffsrichtlinie für Sie generieren lassen, generiert es die folgende Richtlinie. Weitere Informationen zu Zugriffseinträgen finden Sie unter Erlauben des Zugriffs von IAM-Rollen oder Benutzern auf Kubernetes im 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",
            "nonResourceURLs": [
                "/metrics"
            ],
            "verbs": [
                "get"
            ]
        }
    ]
}
```

Manuelles Konfigurieren von Amazon EKS für den Scraper-Zugriff

Wenn Sie es vorziehen, den `aws-auth ConfigMap` Zugriff auf Ihren Kubernetes-Cluster zu kontrollieren, können Sie Amazon Managed Service for Prometheus Scrapern trotzdem 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**
>
> Weitere Informationen zu `ConfigMap` und zum Zugriff auf Einträge finden Sie unter Erlauben des Zugriffs von IAM-Rollen oder Benutzern auf Kubernetes im Amazon EKS-Benutzerhandbuch.

Dieses Verfahren verwendet `kubectl` und die AWS CLI. 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"]
---
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 die AWS API oder die verwenden AWS CLI , um die Scraper in Ihrem Konto aufzulisten oder zu löschen.

> **Note**
>
> Stellen Sie sicher, dass Sie die neueste Version des AWS CLI oder SDK verwenden. Die neueste Version bietet Ihnen die neuesten Features und Funktionen sowie Sicherheitsupdates. Alternativ können Sie AWS Cloudshell verwenden, das automatisch ein stets up-to-date verfügbares Befehlszeilenerlebnis bietet.

Verwenden Sie den ListScrapersAPI-Vorgang, um alle Scraper in Ihrem Konto aufzulisten.

Rufen Sie alternativ mit dem folgenden AWS CLI Befehl auf:

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

Um einen Scraper zu löschen, suchen Sie mithilfe der `ListScrapers` Operation den `scraperId` für den Scraper, den Sie löschen möchten, und verwenden Sie dann den DeleteScraperVorgang, um ihn zu löschen.

Rufen Sie alternativ mit dem folgenden Befehl auf AWS CLI:

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

Scraper-Konfiguration

Mit einer Prometheus-kompatiblen Scraper-Konfiguration können Sie steuern, wie Ihr Scraper Metriken erkennt und sammelt. 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, löschen Sie den Scraper und erstellen Sie ihn mit der neuen Konfiguration neu.

Unterstützte Konfiguration

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

Da Amazon EKS der einzige unterstützte Service ist, ist der einzige unterstützte Service Discovery config (`<*_sd_config>`) der`<kubernetes_sd_config>`.

Die vollständige Liste der zulässigen 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.

```
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
```

Die folgenden Einschränkungen gelten nur für AWS verwaltete Collectors:

- 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.
- Autorisierung — Lassen Sie diese Option aus, wenn keine Autorisierung erforderlich ist. Wenn sie benötigt wird, muss die Autorisierung vorhanden sein `Bearer` und auf die Datei `/var/run/secrets/kubernetes.io/serviceaccount/token` verweisen. Mit anderen Worten, wenn er verwendet wird, muss der Autorisierungsabschnitt 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?

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, müssen Sie wahrscheinlich die Scrape-Konfiguration anpassen. Die Erkennung muss `relabel_config` möglicherweise angepasst werden, oder es ist möglich, dass ein Problem mit der für die Erkennung `role` verwendeten Methode vorliegt.

- up ist vorhanden, aber immer 0 – Wenn up vorhanden, aber 0, dann kann der Kollektor den Endpunkt ermitteln, aber keine Prometheus-kompatiblen Metriken finden.

  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`), der Endpunkt oder der Port, den Sie verwenden. Sie können auch überprüfen, ob der Endpunkt mit einer gültigen `200` Antwort antwortet und dem Prometheus-Format folgt. Schließlich darf der Hauptteil der Antwort nicht größer als die maximal zulässige Größe sein. (Informationen zu den Beschränkungen für 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 Metriken-Endpunkte überprüfen, die gescrapt werden, indem Sie die Anzahl der up Metriken mithilfe von. `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.
- Konto – Ihr EKS-Cluster, Ihr Managed Scraper und Ihr Workspace in Amazon Managed Service für Prometheus müssen sich alle im selben AWS-Konto 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.

Was sind Prometheus-kompatible Metriken?

Um Prometheus-Metriken aus Ihren Anwendungen und Ihrer Infrastruktur für die Verwendung in Amazon Managed Service für Prometheus zu extrahieren, müssen sie Prometheus-kompatible Metriken von Prometheus-kompatiblen `/metrics`-Endpunkten instrumentieren und verfügbar machen. 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 auf einen Prometheus-kompatiblen Endpunkt exportiert werden, können Sie diese Metriken automatisch vom Kollektor von Amazon Managed Service für Prometheus auslesen 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 von Prometheus-kompatiblen Metriken aus Ihrem eigenen Code finden Sie unter Exportieren 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 Verwenden eines verwalteten Collectors AWS.

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 OpenTelemetry die Erfassung von Metriken ermöglicht. 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

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

Verwendung AWS PrivateLink mit Amazon Managed Service für Prometheus

Der Netzwerkverkehr für die Aufnahme der Metriken in Amazon Managed Service for Prometheus kann über einen öffentlichen Internet-Endpunkt 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 for Prometheus mit Schnittstellen-VPC-Endpunkten

Authentifizierung und Autorisierung

AWS Identity and Access Management (IAM) ist ein Webservice, mit dem Sie den Zugriff auf AWS Ressourcen sicher kontrollieren können. 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.

AWS Distro for OpenTelemetry als Collector verwenden

In den folgenden Themen werden verschiedene Möglichkeiten beschrieben, AWS Distro for OpenTelemetry als Collector für Ihre Metriken einzurichten.

Themen

- Richten Sie die Erfassung von Metriken mit AWS Distro for Open Telemetry auf einem Amazon Elastic Kubernetes Service Service-Cluster ein
- Richten Sie die Erfassung von Metriken aus Amazon ECS mit AWS Distro for Open Telemetry ein

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

Richten Sie die Erfassung von Metriken mit AWS Distro for Open Telemetry auf einem Amazon Elastic Kubernetes Service Service-Cluster ein

In diesem Abschnitt wird beschrieben, wie Sie den AWS Distro for OpenTelemetry (ADOT) Collector so konfigurieren, dass er aus einer mit Prometheus instrumentierten Anwendung scrapt und die Metriken an Amazon Managed Service for Prometheus sendet. Weitere Informationen zum ADOT Collector finden Sie unter Distro for.AWS OpenTelemetry

Das Sammeln von Prometheus-Metriken mit ADOT umfasst drei OpenTelemetry Komponenten: den Prometheus Receiver, den Prometheus Remote Write Exporter und die Sigv4 Authentication Extension.

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 Export von Daten werden mit Sigv4, dem AWS Protokoll für sichere Authentifizierung, mit der AWS Sigv4 Authentication Extension 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 Kette der AWS Standardanmeldedaten. Weitere Informationen finden Sie unter Konfiguration des AWS SDK for 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 Richten Sie Servicerollen für die Erfassung von Metriken aus Amazon EKS-Clustern ein befolgen.

   Der ADOT Kollektor verwendet diese Rolle, wenn er Metriken erfasst und exportiert.
2. Bearbeiten Sie als Nächstes die Vertrauensrichtlinie. Öffnen Sie die IAM-Konsole unter https://console.aws.amazon.com/iam/.
3. Wählen Sie im linken Navigationsbereich Rollen aus und suchen Sie nach den Rollen 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:

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

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

```
{
    "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 unter aws-otel-community.

   Führen Sie anschließend folgende Befehle aus.

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

2. Push dieses Image in eine Registry 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 außerdem YOUR_ACCOUNT_ID im Bereich 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 über die Befehlszeile mit AWS Sigv4-Authentifizierung senden. Sie müssen also lokal über AWS Anmeldeinformationen mit den richtigen Berechtigungen für Abfragen von Amazon Managed Service for Prometheus verfügen. Anweisungen zur Installation finden Sie unter awscurl`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 mithilfe von AWS Anmeldeinformationen aus der standardmäßigen Anmeldeinformationskette signiert wurde. AWS Weitere Informationen finden Sie unter Konfigurieren der AWS SDK for Go. Sie müssen den Service als `aps` angeben.

Unabhängig von der Bereitstellungsmethode muss der ADOT-Collector Zugriff auf eine der aufgelisteten Optionen in der Kette der AWS Standardanmeldedaten haben. Die Sigv4-Authentifizierungserweiterung hängt von der ab AWS SDK for Go und verwendet sie, um Anmeldeinformationen abzurufen und sich zu authentifizieren. 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 AWS Distro for Open Telemetry ein

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 aufnehmen. 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.

Definieren Sie ein benutzerdefiniertes ADOT-Kollektor-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 die Werte `endpoint` und `region`. 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 finden Sie unter Authenticator — Sigv4 `sigv4auth` 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]
```

Übertragen Sie Ihr ADOT Kollektor-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 *`my-region`* durch Ihren `region` Wert.

```
# 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
```

Erstellen Sie eine Amazon-ECS-Aufgabendefinition zum Scrapen von Amazon Managed Service für Prometheus

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"
}
```

Weisen Sie einer IAM-Rolle für Ihre Aufgabe die AWS verwaltete Richtlinie **`AmazonPrometheusRemoteWriteAccess`** zu.

Um die gesammelten Metriken an Amazon Managed Service for Prometheus zu senden, muss Ihre Amazon ECS-Aufgabe über die richtigen Berechtigungen verfügen, um die AWS API-Operationen für Sie aufzurufen. 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.

Zeigen Sie Ihre Metriken in Amazon Managed Grafana an

> **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 die IAM-Konsole unter https://console.aws.amazon.com/iam/.

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 als Richtlinienname AmazonPrometheusRemoteWriteAccessdie Option Richtlinie anhängen aus. Wählen Sie Weiter: Tags aus.
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 der Rolle gestartet haben, können Sie eine Demo-Anwendung ausführen, um zu sehen, wie sie funktioniert.

Um eine Demo-Anwendung 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. Ersetzen Sie für `url` *my-region* durch Ihren Region-Wert und *my-workspace-id*durch die Workspace-ID, die Amazon Managed Service for Prometheus für Sie generiert hat. Ersetzen Sie für `region` *my-region* durch Ihren 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

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

> **Warning**
>
> Vermeiden Sie es, Prometheus Scrape-Endpunkte dem öffentlichen Internet auszusetzen, indem Sie Sicherheits-Features aktivieren.

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 benötigen einen Amazon-EKS-Cluster, von dem der neue Prometheus-Server Metriken sammelt
- 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. Ersetzen Sie *prometheus-namespace* durch den Namen, den Sie für diesen Namespace wünschen.

```
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 Richten Sie Servicerollen für die Erfassung von Metriken aus Amazon EKS-Clustern ein, 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-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, den Sie in erstellt haben. amp-iamproxy-ingest-roleRichten Sie Servicerollen für die Erfassung von Metriken aus Amazon EKS-Clustern ein
   - Ersetzen Sie *`WORKSPACE_ID`* durch die ID Ihres Workspace in Amazon Managed Service für Prometheus.
   - Ersetzen Sie *`REGION`* durch die Region Ihres Workspace in Amazon Managed Service für Prometheus.

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

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

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

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

```
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 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 Richten Sie Servicerollen für die Erfassung von Metriken aus Amazon EKS-Clustern ein, 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:

   - `${IAM_PROXY_PROMETHEUS_ROLE_ARN}`Ersetzen Sie durch den ARN des amp-iamproxy-ingest-role, in dem Sie es erstellt habenSchritt 1: Einrichten von IAM-Rollen für Servicekonten. Der Rollen-ARN sollte das Format `arn:aws:iam::`*`your account ID`*`:role/amp-iamproxy-ingest-role` haben.
   - Ersetzen Sie `${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 Signaturprozess von AWS Signature Version 4 (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.

`${SERVICE_ACCOUNT_IAM_INGEST_ROLE_ARN}`Ersetzen Sie durch den ARN des amp-iamproxy-ingest-role, in dem Sie es erstellt habenSchritt 1: Einrichten von IAM-Rollen für Servicekonten. 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 Namespace.

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"
    {"status":"success","data":{"resultType":"vector","result":[{"metric":
{"__name__":"prometheus_api_remote_read_queries","instance":"localhost:9090","job":"prometheus"
[1648461236.419,"0"]}]}}21
```

Einrichten von Amazon Managed Service für Prometheus für Hochverfügbarkeitsdaten

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 Regionsübergreifende Verfügbarkeit.

Themen

- Deduplizierung von Hochverfügbarkeitsmetriken, die an Amazon Managed Service für Prometheus gesendet werden
- Senden Sie mit Prometheus Hochverfügbarkeitsdaten an Amazon Managed Service für Prometheus
- Hochverfügbarkeitsdaten an Amazon Managed Service für Prometheus mit dem Prometheus Operator senden
- Senden Sie mit AWS Distro for Open Telemetry Daten mit hoher Verfügbarkeit an Amazon Managed Service for Prometheus
- Mit dem Prometheus-Community-Helm-Chart Hochverfügbarkeitsdaten an Amazon Managed Service für Prometheus senden
- Fragenforum: Hochverfügbarkeitskonfiguration
- Regionsübergreifende Verfügbarkeit

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 der Amazon Managed Service for Prometheus-Deduplizierung führen. Weitere Informationen finden Sie in den häufig gestellten Fragen zur Hochverfügbarkeit.

In den folgenden Themen wird gezeigt, wie Daten gesendet `cluster` und die `__replica__` Labels hinzugefügt werden, 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 der Amazon Managed Service for Prometheus-Deduplizierung führen. Weitere Informationen finden Sie in den häufig gestellten Fragen zur Hochverfügbarkeit.

Hochverfügbarkeitsdaten an Amazon Managed Service für Prometheus mit dem Prometheus Operator senden

Um eine Hochverfügbarkeitskonfiguration mit dem Prometheus Operator einzurichten, müssen Sie externe Etiketten auf allen Instances einer Hochverfügbarkeitsgruppe anbringen, damit Amazon Managed Service für 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 einem Label aus. `cluster` Dies kann zu Problemen mit der Amazon Managed Service for Prometheus-Deduplizierung führen. Weitere Informationen finden Sie in den häufig gestellten Fragen zur Hochverfügbarkeit.

Senden Sie mit AWS Distro for Open Telemetry Daten mit hoher Verfügbarkeit an Amazon Managed Service for Prometheus

AWS Distro for Open Telemetry (ADOT) ist eine sichere und produktionsreife 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 Distro for Open Telemetry. AWS

Um ADOT mit einer Hochverfügbarkeitskonfiguration einzurichten, müssen Sie ein ADOT-Collector-Container-Image konfigurieren und die externen Labels `cluster` auf den AWS Prometheus `__replica__` Remote Write Exporter anwenden. 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 dem 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 Amazon Managed Service for Prometheus-Deduplizierung 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 einem Label aus. `cluster` Dies kann zu Problemen mit der Amazon Managed Service for Prometheus-Deduplizierung führen. Weitere Informationen finden Sie in den häufig gestellten Fragen zur Hochverfügbarkeit.

Fragenforum: Hochverfügbarkeitskonfiguration

Sollte ich den Wert __replica__ in ein anderes Label aufnehmen, um die Probenpunkte nachzuverfolgen?

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 ein eigenes Cluster-Label zu haben und dedupliziert meine Metriken nicht. Wie lässt sich dieses Problem beheben?

In Kubernetes 1.28 `apiserver_storage_size_bytes` wurde eine neue Metrik mit einem Label eingeführt. `cluster` Dies kann zu Problemen mit der Deduplizierung in Amazon Managed Service for Prometheus führen, die vom Label abhängen. `cluster` In Kubernetes 1.3 wurde das Label umbenannt in `storage-cluster_id` (es wurde auch in späteren Patches von 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` Etikett auf Ihrer `apiserver_storage_size_bytes` Metrik neu etikettieren, bevor Sie sie in Amazon Managed Service for Prometheus aufnehmen.

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

Regionsübergreifende Verfügbarkeit

Um Ihren Daten eine regionsübergreifende Verfügbarkeit zu verleihen, können Sie Metriken AWS an mehrere Workspaces in 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. Sie können einen Service wie Grafana verwenden, um die Metriken abzufragen, oder Sie können die APIs von Amazon Managed Service für Prometheus verwenden.

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.

Themen

Sicherung Ihrer metrischen Abfragen

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

Verwendung 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 Wenn Sie diese Option verwenden 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 for 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 Integration von Alerts mit 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 for Prometheus.

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

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).

Anweisungen zum Einrichten eines eigenständigen Grafana mithilfe der tar.gz- oder Zip-Datei finden Sie in der Grafana-Dokumentation unter Grafana installieren. Wenn Sie ein neues eigenständiges Grafana installieren, werden Sie zur Eingabe von Benutzername und Passwort aufgefordert. Der Standardwert ist **`admin/admin`**. Sie werden aufgefordert, das Passwort zu ändern, nachdem Sie sich zum ersten Mal angemeldet haben. Weitere Informationen finden Sie unter Erste Schritte mit Grafana in der Grafana-Dokumentation.

Um Ihre Grafana-Version zu überprüfen, führen Sie den folgenden Befehl aus.

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

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

AWS SigV4 einrichten

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
```

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 wie https://aps-workspaces.us-west-2.amazonaws.com/workspaces/ws-1234a5b6-78cd-901e-2fgh-3i45j6k178l9 aussehen.
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.

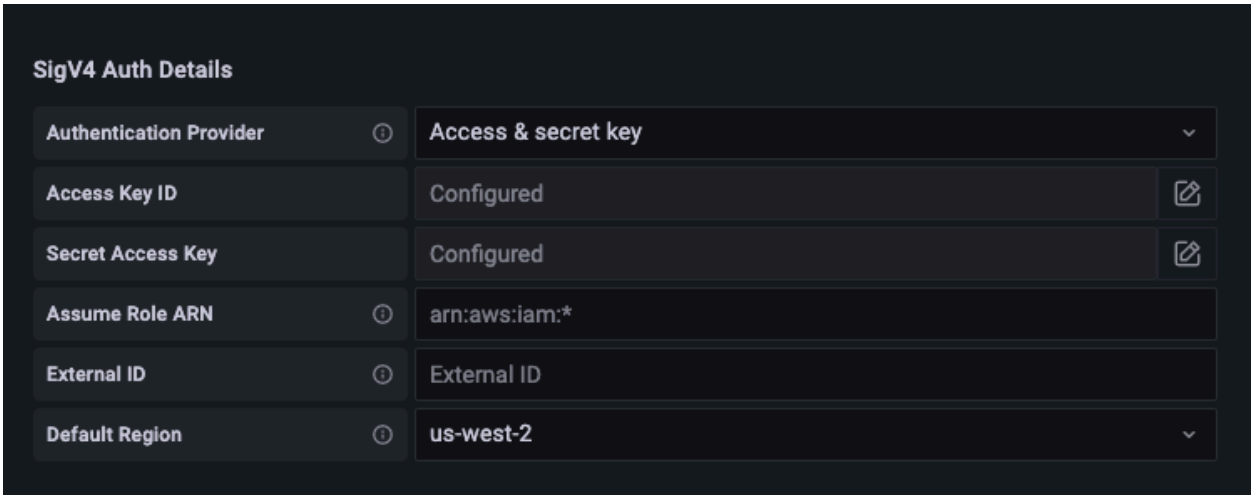
     d. Lassen Sie die Felder ARN-Rolle übernehmen und Externe ID leer.

     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.

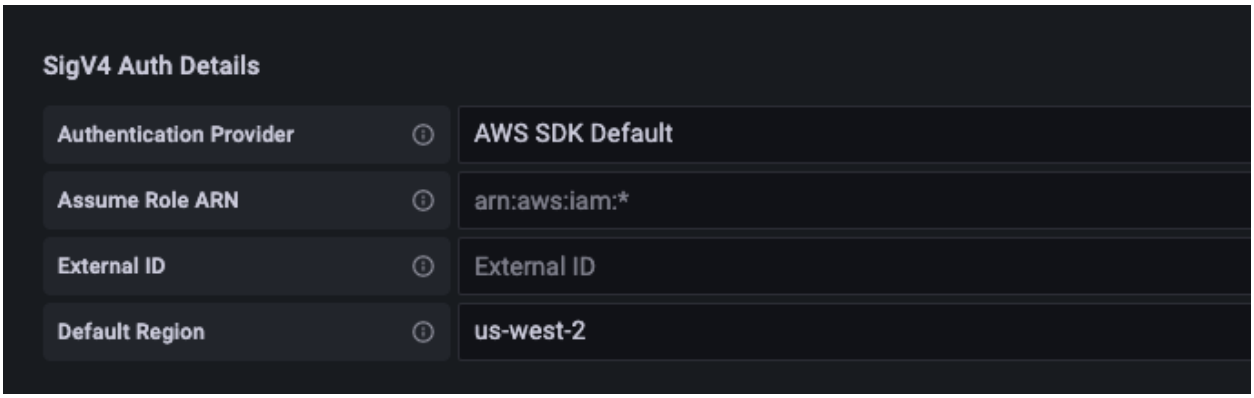
  b. Lassen Sie die Felder ARN-Rolle übernehmen und Externe ID leer.

  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

  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
```

Fehlerbehebung, wenn „Speichern & Testen“ 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 der Richtlinie entsprechen. AmazonPrometheusQueryAccess 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 dieaps:QueryMetrics, aps:GetMetricMetadataaps:GetSeries, und Berechtigungen verfügt. aps:GetLabels Weitere Informationen finden Sie unter IAM-Berechtigungen und Richtlinien.

Richten Sie SigV4 ein AWS

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. Ersetzen Sie *Account-ID* 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.

   ```
   {
     "Version": "2012-10-17",
     "Statement": [
       {
         "Effect": "Allow",
         "Principal": {
           "Federated": "arn:aws:iam::account-id:oidc-provider/
   oidc.eks.aws_region.amazonaws.com/id/openid"
         },
         "Action": "sts:AssumeRoleWithWebIdentity",
         "Condition": {
           "StringEquals": {
             "oidc.eks.region.amazonaws.com/id/openid:sub": [
               "system:serviceaccount:aws-amp:amp-iamproxy-query-service-account",
               "system:serviceaccount:grafana-namespace:grafana-service-account-name"
             ]
   ```

```
        }
      }
    }
  ]
}
```

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 Prometheus-kompatiblen APIs

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

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 den AWS SigV4-Proxy kann mit `awscurl` durchgeführt werden. Das folgende Thema Verwenden von awscurl zur Abfrage von Prometheus-kompatiblen APIs führt Sie durch die Verwendung von `awscurl` zur Einrichtung von AWS SigV4.

Verwenden von awscurl zur Abfrage von Prometheus-kompatiblen APIs

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 `awscurl` Beispielabfrage. Ersetzen Sie die Eingaben *`Region`*, *`Workspace-ID`* und *`QUERY`* 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. Den Zugriffsschlüssel und den geheimen Schlüssel für die Rolle finden Sie unter https://console.aws.amazon.com/iam/.

  Beispielsweise:

  ```
  $ 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. Beispielsweise:

  ```
  $ 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"
```

Statistikinformationen in der Abfrage-API-Antwort abfragen

Die Preisgestaltung für Abfragen basiert auf der Gesamtzahl der in einem Monat verarbeiteten Abfrageproben aus ausgeführten Abfragen. Die Abfrageantwort für eine `query` oder eine `queryRange` API umfasst die statistischen Daten über die verarbeiteten Abfrageproben. Wenn der Abfrageparameter `stats=all` in der Anfrage gesendet wird, wird ein `samples`-Objekt im `stats`-Objekt erstellt und die `stats`-Daten werden in der Antwort zurückgegeben.

Das `samples` Objekt hat die folgenden Attribute:

Attribut	Beschreibung
`totalQueryableSamples`	Gesamtzahl der verarbeiteten Abfragebeispiele. Dies sind die Informationen, die für die Abrechnung verwendet werden sollen.
`totalQueryableSamp lesPerStep`	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
                    ]
                ]
            }
        }
    }
}
```

Aufzeichnungs- und Alarmregeln

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 die Benachrichtigung anschließend 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

- Erforderliche IAM-Berechtigungen

- Erstellen einer Regeldatei
- Hochladen einer Regelkonfigurationsdatei auf Amazon Managed Service für Prometheus
- Eine Regelkonfigurationsdatei bearbeiten
- Fehlersuche-Ruler

Erforderliche IAM-Berechtigungen

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.

```
{
    "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.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "aps:ListRules",
                "aps:ListTagsForResource",
                "aps:GetLabels",
                "aps:CreateRuleGroupsNamespace",
                "aps:ListRuleGroupsNamespaces",
                "aps:DescribeRuleGroupsNamespace",
                "aps:PutRuleGroupsNamespace",
                "aps:DeleteRuleGroupsNamespace"
            ],
            "Resource": [
                "arn:aws:aps:*:<account>:workspace/*",
                "arn:aws:aps:<region>:<account>:rulegroupnamespace/<workspace-
id>/<namespace-name>"
            ]
        }
    ]
}
```

Erstellen einer 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 für Prometheus hat dasselbe Format wie eine Regeldatei in der eigenständigen Version von Prometheus. Weitere Informationen finden Sie unter Definieren von Aufzeichnungsregeln und Alarmregeln.

Im Folgenden wird ein Basisbeispiel für eine Regeldatei dargestellt:

```
groups:
```

```
  - name: test
    rules:
    - record: metric:recording_rule
      expr: avg(rate(container_cpu_usage_seconds_total[5m]))
  - name: alert-test
    rules:
    - alert: metric:alerting_rule
      expr: avg(rate(container_cpu_usage_seconds_total[5m])) > 0
      for: 2m
```

Weitere Beispiele für Alarmregeln finden Sie unter Beispiele für Alarmregeln.

> **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 unterHochladen einer Regelkonfigurationsdatei auf Amazon Managed Service für Prometheus.

Hochladen einer Regelkonfigurationsdatei auf Amazon Managed Service für Prometheus

Sobald Sie wissen, welche Änderungen Sie an der Regelkonfigurationsdatei vornehmen möchten, können Sie sie entweder in der Konsole bearbeiten oder eine Ersatzdatei 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 Konsole von Amazon Managed Service für Prometheus 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 den zu verwenden 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.

Eine Regelkonfigurationsdatei bearbeiten

Sie können entweder eine neue Regeldatei hochladen, um eine 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 Konsole von Amazon Managed Service für Prometheus 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 die AWS CLI zur Bearbeitung einer Regelkonfigurationsdatei zu verwenden

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.

Fehlersuche-Ruler

Mit CloudWatch Logs 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.

Alert Manager

Wenn die von Amazon Managed Service für Prometheus ausgeführten Alarmregeln ausgelöst werden, verarbeitet der Alert Manager die gesendeten Alarme. Er dedupliziert, gruppiert und leitet die Alarme an nachgelagerte 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

Das eigenständige Prometheus unterstützt das Vorlagensystem, wobei separate Vorlagendateien verwendet werden. 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

Erforderliche IAM-Berechtigungen

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.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "aps: CreateAlertManagerDefinition",
                "aps: DescribeAlertManagerSilence",
                "aps: DescribeAlertManagerDefinition",
                "aps: PutAlertManagerDefinition",
                "aps: DeleteAlertManagerDefinition",
                "aps: ListAlerts",
                "aps: ListRules",
                "aps: ListAlertManagerReceivers",
                "aps: ListAlertManagerSilences",
```

```
                "aps: ListAlertManagerAlerts",
                "aps: ListAlertManagerAlertGroups",
                "aps: GetAlertManagerStatus",
                "aps: GetAlertManagerSilence",
                "aps: PutAlertManagerSilences",
                "aps: DeleteAlertManagerSilence",
                "aps: CreateAlertManagerAlerts"
            ],
            "Resource": "*"
        }
    ]
}
```

Eine Alert-Manager-Konfigurationsdatei erstellen

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. Hochladen Ihrer Alert-Manager-Konfigurationsdatei nach Amazon Managed Service für Prometheus

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 }}
```

Einrichten Ihres Alarmempfängers

Amazon Simple Notification Service (Amazon SNS) ist der einzige Alarmempfänger, der derzeit in Amazon Managed Service für Prometheus unterstützt wird. Weitere Informationen finden Sie unter Was ist Amazon SNS?.

Themen

- (Optional) Erstellen eines neuen Amazon-SNS-Themas
- Amazon Managed Service für Prometheus die Erlaubnis erteilen, Nachrichten an Ihr Amazon-SNS-Thema zu senden

- Angabe Ihres Amazon-SNS-Themas in der Alert-Manager-Konfigurationsdatei
- (Optional) Konfiguration des Alert Managers für die Ausgabe von JSON an Amazon SNS
- (Optional) das Senden von Amazon SNS an andere Ziele
- Regeln für die Validierung und Kürzung von SNS-Empfängernachrichten

(Optional) Erstellen eines neuen Amazon-SNS-Themas

Sie können ein vorhandenes SNS-Thema verwenden oder ein neues Thema 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.

Amazon Managed Service für Prometheus die Erlaubnis erteilen, Nachrichten an Ihr Amazon-SNS-Thema zu senden

Sie müssen Amazon Managed Service für Prometheus die Erlaubnis erteilen, Nachrichten an Ihr Amazon-SNS-Thema zu senden. Die folgende Grundsatzerklärung enthält eine `Condition` Erklärung, mit der das Sicherheitsproblem „Confused Deputy“ vermieden werden soll. 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-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 SNS-Thema Service Side Encryption (SSE) aktiviert ist, müssen Sie Ihrer KMS-Schlüsselrichtlinie im `"Action"` Block die folgenden Berechtigungen hinzufügen. Weitere Informationen finden Sie unter AWS KMS-Berechtigungen für SNS-Themen.

```
    kms:GenerateDataKey
    kms:Decrypt
```

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.

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. In AWS kann ein dienstübergreifendes Identitätswechsels zum Problem des verwirrten Stellvertreters führen. Ein dienstübergreifender Identitätswechsel kann auftreten, wenn ein Dienst (der Anruf-Dienst) einen anderen Dienst anruft (den aufgerufenen Dienst). Der Anruf-Dienst kann so manipuliert werden, dass er seine Berechtigungen verwendet, um auf die Ressourcen eines anderen Kunden zu reagieren, auf die er sonst nicht zugreifen dürfte. Um dies zu verhindern, AWS bietet Tools, mit denen Sie Ihre Daten für alle Dienste mit Dienstprinzipalen schützen können, denen Zugriff auf Ressourcen in Ihrem Konto gewährt wurde.

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. z. B. `arn:aws:servicename::123456789012:*`.

Das folgende Beispiel in Amazon Managed Service für Prometheus die Erlaubnis erteilen, Nachrichten an Ihr Amazon-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.

Angabe Ihres Amazon-SNS-Themas in der Alert-Manager-Konfigurationsdatei

Jetzt können Sie Ihren Amazon-SNS-Empfänger zu Ihrer Alert-Manager-Konfiguration hinzufügen. Dazu 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: region
      topic_arn: ARN_of_SNS_topic
      subject: somesubject
      attributes:
        key: somekey
        value: somevalue
```

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 (Optional) Konfiguration des Alert Managers für die Ausgabe von JSON an Amazon SNS in diesem Handbuch.

Jetzt müssen Sie Ihre Alert-Manager-Konfigurationsdatei nach Amazon Managed Service für Prometheus hochladen. Weitere Informationen finden Sie unter Hochladen Ihrer Alert-Manager-Konfigurationsdatei nach Amazon Managed Service für Prometheus.

(Optional) Konfiguration des Alert Managers für die Ausgabe von JSON an Amazon SNS

Sie können den Alert Manager so konfigurieren, dass er Warnmeldungen im JSON-Format sendet, sodass sie nachgelagert von Amazon SNS in AWS Lambda oder in Webhook-Empfangsendpunkten verarbeitet werden können. Die mit dem Alert Manager von Amazon Managed Service für Prometheus bereitgestellte Standardvorlage gibt die Nachrichtennutzdaten in einem Textlistenformat aus, das möglicherweise nicht einfach zu analysieren ist. 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 Eine Alert-Manager-Konfigurationsdatei erstellen.
- das Hochladen Ihrer Konfigurationsdatei, siehe Hochladen Ihrer Alert-Manager-Konfigurationsdatei nach Amazon Managed Service für Prometheus.

(Optional) das Senden von Amazon SNS an andere Ziele

Derzeit kann Amazon Managed Service für Prometheus Alarmmeldungen nur direkt an Amazon SNS senden. Sie können Amazon SNS so konfigurieren, dass diese Nachrichten an andere Ziele wie E-Mail, Webhook, Slack und gesendet werden. 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

Regeln für die Validierung und Kürzung von SNS-Empfängernachrichten

SNS-Nachrichten werden vom SNS-Empfänger auf der Grundlage der folgenden Regeln validiert, gekürzt oder gegebenenfalls geändert:

- Die Nachricht enthält Nicht-UTF-Zeichen.
  - Die Nachricht wird durch „Fehler - keine gültige UTF-8-kodierte Zeichenfolge“ ersetzt.
  - Ein Nachrichtenattribut wird mit dem Schlüssel „gekürzt“ und dem Wert „wahr“ hinzugefügt
  - Es wird ein Nachrichtenattribut mit dem Schlüssel „geändert“ und dem Wert „Nachricht: Fehler – kein gültiger UTF-8-codierter String“ hinzugefügt.
- Die Nachricht ist leer.
  - Die Nachricht wird durch „Fehler – Nachricht sollte nicht leer sein“ ersetzt.
  - Es wird ein Nachrichtenattribut mit dem Schlüssel „geändert“ und dem Wert „Nachricht: Fehler – Nachricht sollte nicht leer sein“ hinzugefügt.
- Die Nachricht wurde gekürzt.
  - Die Nachricht wird den gekürzten Inhalt haben.
  - Ein Nachrichtenattribut wird mit dem Schlüssel „gekürzt“ und dem Wert „wahr“ hinzugefügt
  - Es wird ein Nachrichtenattribut mit dem Schlüssel „geändert“ und dem Wert „Nachricht: Fehler – Nachricht wurde um *X* KB gekürzt, weil sie die Größenbeschränkung von 256 KB überschreitet“ hinzugefügt.

- Der Betreff ist kein ASCII.
  - Der Betreff wird durch „Fehler - enthält nicht druckbare ASCII-Zeichen“ ersetzt.
  - 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 *X* Zeichen gekürzt, weil er die Größenbeschränkung von 100 Zeichen überschreitet“ hinzugefügt.
- Das Nachrichtenattribut hat einen ungültigen Schlüssel/Wert.
  - Ungültiges Nachrichtenattribut wird entfernt.
  - Es wird ein Nachrichtenattribut mit dem Schlüssel „modifiziert“ und dem Wert "hinzugefügtMessageAttribute: Fehler — *X* der Nachrichtenattribute wurden entfernt, weil sie ungültig sind oder.“ MessageAttributeKey MessageAttributeValue
- Das Nachrichtenattribut wurde gekürzt.
  - Zusätzliche Nachrichtenattribute werden entfernt.
  - Es wird ein Nachrichtenattribut mit dem Schlüssel „modifiziert“ und dem Wert "hinzugefügtMessageAttribute: Fehler — *X* der Nachrichtenattribute wurden entfernt, weil es die Größenbeschränkung von 256 KB überschreitet.

Hochladen Ihrer Alert-Manager-Konfigurationsdatei nach Amazon Managed Service für Prometheus

Sobald Sie wissen, welche Änderungen Sie an der Alert Manager-Konfigurationsdatei vornehmen möchten, können Sie sie entweder in der Konsole bearbeiten oder eine Ersatzdatei mit der Konsole oder AWS CLI hochladen.

> **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 Konsole von Amazon Managed Service für Prometheus 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 fileb://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.

Integration von Alerts mit Amazon Managed Grafana oder Open-Source-Grafana

Alert-Regeln, die Sie in Alert Manager innerhalb von Amazon Managed Service for Prometheus erstellt haben, können in Amazon Managed Grafana und Grafana weitergeleitet und angezeigt werden, wodurch Ihre Alert-Regeln und Alerts in einer einzigen Umgebung vereinheitlicht werden. In Amazon Managed Grafana können Sie Ihre Alert-Regeln und die generierten Alerts anzeigen.

Voraussetzungen

Bevor Sie mit der Integration von Amazon Managed Service for 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 AWS-Konto und IAM-Anmeldeinformationen finden Sie unter Einrichtung.
- Sie müssen über einen Amazon Managed Service for Prometheus Workspace verfügen und Daten in diesen aufnehmen. Informationen zum Einrichten eines neuen Workspace finden Sie unter Einen Workspace erstellen. Sie sollten auch mit den Prometheus-Konzepten wie Alert Manager und Ruler vertraut sein. Weitere Informationen zu diesen Themen finden Sie in der Prometheus-Dokumentation.

- Sie haben bereits eine Alert Manager-Konfiguration und eine Regeldatei in Amazon Managed Service for Prometheus konfiguriert. Weitere Informationen über Alert Manager in Amazon Managed Service for Prometheus finden Sie in Alert Manager. Weitere Informationen zu Regeln finden Sie unter Aufzeichnungs- und Alarmregeln.
- Sie müssen entweder Amazon Managed Grafana eingerichtet oder die Open-Source-Version von Grafana ausgeführt haben.
  - Wenn Sie Amazon Managed Grafana verwenden, müssen Sie Grafana-Alerting verwenden. Weitere Informationen finden Sie unter Migration älterer Dashboard-Alerts zu Grafana-Alerting.
  - 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-Alerting) aktivieren und möglicherweise einen Sigv4-Proxy einrichten, um Aufrufe von Grafana an Amazon Managed Service for 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 Amazon Managed Service for Prometheus Instance bereits Regeln und Alerts eingerichtet haben, erfolgt die Konfiguration zur Verwendung von Amazon Managed Grafana als Dashboard für diese Alerts vollständig in Amazon Managed Grafana.

So konfigurieren Sie Amazon Managed Grafana als Ihr Alerts-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 Fügen Sie die Prometheus-Datenquelle in Grafana hinzu.
4. Wählen Sie in der Prometheus-Datenquelle die Option Alerts über die Alert Manager-Benutzeroberfläche verwalten aus.
5. Kehren Sie zur Datenquellen-Oberfläche zurück.
6. Erstellen Sie eine neue Alert Manager-Datenquelle.
7. Fügen Sie auf der Konfigurationsseite für die Alert Manager-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. Zum Beispiel *`https://aps-workspaces.us-east1.amazonaws.com/workspaces/ws-example-1234-5678-abcd-xyz00000001/alertmanager.`*
   - Aktivieren Sie unter Auth die Option SigV4Auth. Dadurch wird Grafana angewiesen, die AWSAuthentifizierung 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 Amazon Managed Service for Prometheus Alerts sollten jetzt so konfiguriert sein, dass sie mit Ihrer Grafana-Instance funktionieren. Vergewissern Sie sich, dass Sie alle Alert-Regeln, Alert-Gruppen (einschließlich aktiver Alerts) und Silences aus Ihrer Amazon Managed Service for Prometheus-Instance auf der Grafana-Alerting-Seite sehen können.

Fehlerbehebung bei Alert Manager

Mit CloudWatch Logs 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 leerem Inhalt

Wenn das Protokoll die folgende Warnung enthält

```
{
   "workspaceId": "ws-efdc5b42-b051-11ec-b123-4567ac120002",
   "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.

Nicht-ASCII-Warnung

Wenn das Protokoll die folgende Warnung enthält

```
{
   "workspaceId": "ws-efdc5b42-b051-11ec-b123-4567ac120002",
   "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.

Ungültige **`key/value`** Warnung

Wenn das Protokoll die folgende Warnung enthält

```
{
   "workspaceId": "ws-efdc5b42-b051-11ec-b123-4567ac120002",
   "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 Schlüssel/Werte ungültig waren.

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-efdc5b42-b051-11ec-b123-4567ac120002",
   "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-efdc5b42-b051-11ec-b123-4567ac120002",
   "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.

Protokollierung und Überwachung

Sie können die Ressourcennutzung Ihres Amazon Managed Service for Prometheus mit den CloudWatch Protokollierungs- und Überwachungsfunktionen von Amazon verwalten.

- Überwachen Sie Amazon Managed Service für Prometheus mit CloudWatch Metriken.
- Mit CloudWatch Logs können Sie die Alert-Manager- und Ruler-Ereignisse von Amazon Managed Service für Prometheus abfragen und anzeigen.

CloudWatch Metriken

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	Ressource nname	CloudWatch Namespace	Beschreibung
ResourceCount	IngestionRate	`AWS/Usage`	Beispielerfassungsrate Einheiten: Anzahl pro Sekunde Gültige Statistiken: Minimum, Maximum, Summe, Durchschnitt
ResourceCount	ActiveSeries	`AWS/Usage`	Anzahl der aktiven Serien pro Workspace Einheiten: Anzahl Gültige Statistiken: Minimum, Maximum, Summe, Durchschnitt

CloudWatch Name der Metrik	Ressource nname	CloudWatch Namespace	Beschreibung
ResourceCount	ActiveAlerts	AWS/Usage	Anzahl der aktiven Alarme pro Workspace Einheiten: Anzahl Gültige Statistiken: Minimum, Maximum, Summe, Durchschnitt
ResourceCount	SizeOfWarnmeldungen	AWS/Usage	Gesamtgröße aller Benachrichtigungen im Workspace, in Byte Einheiten: Byte Gültige Statistiken: Minimum, Maximum, Summe, Durchschnitt
ResourceCount	SuppressedAlerts	AWS/Usage	Anzahl der Alarme im unterdrückten Status pro Workspace. Ein Alarm kann durch Stummschalten oder Unterdrückung unterbunden werden. Einheiten: Anzahl Gültige Statistiken: Minimum, Maximum, Summe, Durchschnitt

CloudWatch Name der Metrik	Ressource nname	CloudWatch Namespace	Beschreibung
ResourceCount	UnprocessedAlerts	AWS/Usage	Anzahl der Alarme im unbearbeiteten Zustand pro Workspace. Eine Warnung befindet sich im unverarbeiteten Zustand, sobald sie von empfangen wurde AlertManager, wartet aber auf die nächste Auswertung der Aggregationsgruppe. Einheiten: Anzahl Gültige Statistiken: Minimum, Maximum, Summe, Durchschnitt
ResourceCount	AllAlerts	AWS/Usage	Anzahl der Alarme in einem beliebigen Zustand pro Workspace. Einheiten: Anzahl Gültige Statistiken: Minimum, Maximum, Summe, Durchschnitt

CloudWatch Name der Metrik	Ressource nname	CloudWatch Namespace	Beschreibung
AlertManagerAlerts Received	-	`AWS/Prometheus`	Gesamtzahl der erfolgreichen Alarme, die vom Alert Manager empfangen wurden Einheiten: Anzahl Gültige Statistiken: Minimum, Maximum, Summe, Durchschnitt
AlertManagerNotifications Failed	-	`AWS/Prometheus`	Anzahl der fehlgeschlagenen Alarme Einheiten: Anzahl Gültige Statistiken: Minimum, Maximum, Summe, Durchschnitt
AlertManagerNotifications Throttled	-	`AWS/Prometheus`	Anzahl der gedrosselten Alarme Einheiten: Anzahl Gültige Statistiken: Minimum, Maximum, Summe, Durchschnitt
Discarded Samples*	-	`AWS/Prometheus`	Anzahl der verworfenen Beispiele nach Grund Einheiten: Anzahl Gültige Statistiken: Minimum, Maximum, Summe, Durchschnitt

CloudWatch Name der Metrik	Ressource nname	CloudWatch Namespace	Beschreibung
RuleEvaluations	-	`AWS/Prometheus`	Gesamtzahl der Regelauswertungen Einheiten: Anzahl Gültige Statistiken: Minimum, Maximum, Summe, Durchschnitt
RuleEvaluationFehlschläge	-	`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

* Einige der Gründe, die dazu führen, dass Beispiele verworfen werden, sind folgende.

Grund	Bedeutung
greater_than_max_sample_age	Verwerfen von Proben, die älter als eine Stunde sind.

Grund	Bedeutung
new-value-for-timestamp	Doppelte Proben werden mit einem anderen Zeitstempel als der zuvor aufgezeichneten gesendet.
per_metric_series_limit	Der Benutzer hat das Limit für die aktive Serie 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.
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.
label_invalid	Ungültiges Etikett angegeben.
duplicate_label_names	Doppelte Labelnamen angegeben.

Note

Eine Metrik, die nicht existiert oder fehlt, entspricht dem Wert dieser Metrik, der 0 ist.

Note

`RuleGroupIterationsMissed`, `RuleEvaluations` und `RuleEvaluationFailures` 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 Nummer von ActiveSeriesin Prometheus einzustellen

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.

CloudWatch Logs

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

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.

```
{
    "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.

Kosten verstehen und optimieren

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 zum Umetikettieren 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.

Um mehr Einblick in die Anfragen zu erhalten, die am meisten zu Ihren Abfragekosten beitragen, können Sie sich an Ihren Support-Ansprechpartner wenden und ein Ticket einreichen. Das Team von Amazon Managed Service für Prometheus kann Ihnen helfen, die Anfragen zu verstehen, die am meisten zu Ihren Kosten beitragen.

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.

Wenn Sie Ihre Aufbewahrungsfrist verkürzen (oder verlängern) möchten, können Sie eine Anfrage zur Änderung der `Retention time for ingested data` Quote stellen.

Wie kann ich die Kosten für meine Warnungsabfrage 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. Alert Manager

Welche Metriken kann ich verwenden, um meine Kosten zu überwachen?

Überwachen Sie `IngestionRate` bei Amazon CloudWatch , um Ihre Aufnahmekosten zu verfolgen. Weitere Informationen zur Überwachung von Amazon Managed Service for Prometheus-Metriken finden Sie unter CloudWatch. CloudWatch Metriken

Kann ich meine Rechnung jederzeit überprüfen?

Darin wird Ihre AWS Nutzung AWS Cost and Usage Report nachverfolgt und geschätzte 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

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-Services

Amazon Managed Service for Prometheus lässt sich in andere AWS Services integrieren. In diesem Abschnitt werden die Integration der Kostenüberwachung mit Amazon Elastic Kubernetes Service (Amazon EKS) (mit Kubecost) und die Verwendung von Terraform-Modulen zur Erstellung einer vollständigen Observability-Lösung für Ihre EKS-Projekte mit AWS Observability Accelerator beschrieben.

Themen

Integration in Amazon EKS-Kostenüberwachung

Amazon Managed Service for 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 for 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 for 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 for Prometheus.

> **Note**
>
> Weitere Informationen zur Verwendung von Kubecost finden Sie unter Kostenüberwachung im Amazon EKS-Benutzerhandbuch.

AWS Observability Accelerator verwenden

AWS bietet Observability-Tools, einschließlich Überwachung, Protokollierung, Alerting und Dashboards, für Ihre Amazon Elastic Kubernetes Service (Amazon EKS) -Projekte. Dazu gehören Amazon Managed Service for Prometheus, Amazon Managed Grafana, AWSDistro für OpenTelemetry und andere Tools. Um Ihnen zu helfen, diese Tools zusammen zu verwenden, bietet AWS Terraform-Module, die Observability mit diesen Services konfigurieren, den sogenannten AWSObservability Accelerator.

AWS Observability Accelerator bietet Beispiele für die Überwachung von Infrastrukturen, NGINX-Bereitstellungen und anderen Szenarien. Dieser Abschnitt enthält ein Beispiel für die Überwachung der Infrastruktur in Ihrem Amazon EKS-Cluster.

Die Terraform-Vorlagen und detaillierte Anweisungen finden Sie auf der GitHub-Seite AWSObservability Accelerator for Terraform. Sie können auch den Blogbeitrag zur AnkündigungAWS von Observability Accelerator lesen.

Voraussetzungen

Um AWS Observability Accelerator verwenden zu können, benötigen Sie einen Amazon-EKS-Cluster, der die folgenden Voraussetzungen erfüllt:

- AWS CLI— wird verwendet, um AWS Funktion von der Befehlszeile aus aufzurufen.
- kubectl — wird verwendet, um Ihren EKS-Cluster von der Befehlszeile aus zu steuern.
- Terraform — wird verwendet, um die Erstellung der Ressourcen für diese Lösung zu automatisieren. Sie müssen den AWS Anbieter mit einer IAM-Rolle eingerichtet haben, die Zugriff auf die Erstellung und Verwaltung von Amazon Managed Service for Prometheus,

Amazon Managed Grafana und IAM innerhalb Ihres AWS Kontos hat. Weitere Informationen zur Konfiguration des AWS Anbieters für Terraform finden Sie unter AWSAnbieter in der Terraform-Dokumentation.

Verwenden Sie das Beispiel für die Infrastrukturüberwachung

AWS Observability Accelerator bietet Beispielvorlagen, die die mitgelieferten Terraform-Module verwenden, um Observability für Ihren Amazon EKS-Cluster einzurichten und zu konfigurieren. Dieses Beispiel zeigt die Verwendung von AWS Observability Accelerator zur Einrichtung der Infrastrukturüberwachung. Weitere Informationen zur Verwendung dieser Vorlage und zu den zusätzlichen Funktionen, die sie enthält, finden Sie auf der GitHub-Seite unter Bestehender Cluster mit der AWS Observability Accelerator Basis und Infrastrukturüberwachung.

Um das Terraform-Modul zur 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/existing-cluster-with-base-and-infra

terraform init
```

3. Erstellen Sie eine neue `terraform.tfvars` Datei wie im folgenden Beispiel. Verwenden Sie die AWS Region und die Cluster-ID für Ihren Amazon-EKS-Cluster.

```
# (mandatory) AWS Region where your resources will be located
aws_region = "eu-west-1"

# (mandatory) EKS Cluster name
eks_cluster_id = "my-eks-cluster"
```

4. Erstellen Sie einen Amazon-Managed Grafana Workspace, wenn Sie noch keinen haben, den Sie verwenden möchten. Informationen zum Erstellen eines neuen Workspace finden Sie unter Erstellen Sie Ihren ersten Workspace im Amazon Managed Grafana-Benutzerhandbuch.

5. Erstellen Sie zwei Variablen für Terraform, um Ihren Grafana-Workspace zu verwenden, indem Sie die folgenden Befehle in der Befehlszeile ausführen. Sie müssen die *grafana-workspace-id* durch die 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 *prometheus-workspace-id* durch Ihre 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 Amazon Managed Service for Prometheus Workspace (sofern Sie sich nicht für die Nutzung eines vorhandenen Workspace entschieden haben).
- Konfiguration, Alerts und Regeln des Alert Managers in Ihrem Prometheus-Arbeitsbereich.
- Neue Amazon Managed Grafana-Datenquelle und Dashboards in Ihrem aktuellen Workspace. Die Datenquelle wird `aws-observability-accelerator` genannt. Die Dashboards werden unter Observability Accelerator Dashboards aufgeführt.
- Ein AWSDistro for OpenTelemetry-Operator der im bereitgestellten Amazon EKS-Cluster eingerichtet wurde, um Metriken an Ihren Amazon Managed Service for Prometheus Workspace zu senden.

Um Ihre neuen Dashboards anzuzeigen, öffnen Sie das entsprechende Dashboard in Ihrem Amazon Managed Grafana Workspace. Weitere Informationen zur Verwendung von Amazon Managed

Grafana finden Sie unter Arbeiten in Ihrem Grafana Workspace im Amazon Managed Grafana-Benutzerhandbuch.

Integration mit AWS Controllern für Kubernetes

Amazon Managed Service for Prometheus ist in AWS Controllers for Kubernetes (ACK) integriert und unterstützt die Verwaltung Ihrer Workspace-, Alert Manager- und Ruler-Ressourcen in Amazon EKS. Sie können AWS Controller für benutzerdefinierte Kubernetes-Ressourcendefinitionen (CRDs) 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 für Prometheus in einem vorhandenen Amazon-EKS-Cluster einrichten.

Sie können auch die Blogbeiträge lesen, in denen AWS Controller für Kubernetes und der ACK-Controller für Amazon Managed Service für Prometheus vorgestellt werden.

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 müssen über vorhandene - AWS-Konto und -Berechtigungen verfügen, 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, *YOUR_CLUSTER_NAME und AWS_REGION durch die richtigen* Werte für 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. Die AWS CLI wird verwendet, um AWS Funktionen über die Befehlszeile 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 for 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 können Sie einen AWS -Controller für Kubernetes installieren und ihn zur Bereitstellung von Amazon Managed Service for Prometheus-Ressourcen verwenden.

Bereitstellen eines Workspace mit AWS Controllern für Kubernetes

Um einen neuen Amazon Managed Service for Prometheus Workspace bereitzustellen, installieren Sie einen Controller für AWS -Controller für Kubernetes und verwenden ihn dann, um den Workspace zu erstellen.

So stellen Sie einen neuen Amazon Managed Service for Prometheus Workspace mit AWS Controllern für Kubernetes bereit

1. Verwenden Sie die folgenden Befehle, um Helm zur Installation des Amazon Managed Service for Prometheus Service Controller zu verwenden. Weitere Informationen finden Sie unter Installieren eines ACK Controllers in der Dokumentation AWS zu Controllern für Kubernetes auf GitHub. Verwenden Sie die richtige *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 | grep '"tag_name":' | cut -d'"' -f4`
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"
```

Mit dem folgenden Befehl können Sie optional überprüfen, ob der AWS Controllers 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 Amazon Managed Service for Prometheus Konsole 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 und AlertManager , die Sie als Nächstes erstellen, mit der folgenden Konfiguration.

   Speichern Sie diese Konfiguration unter `rulegroup.yaml`. Ersetzen Sie *`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`. Ersetzen Sie *WORKSPACE-ID* durch die Workspace ID aus dem vorherigen Schritt. Ersetzen Sie *TOPIC-ARN* durch den ARN für das Amazon SNS-Thema, an das Benachrichtigungen gesendet werden sollen, und *REGION* durch die , die AWS-Region Sie verwenden. Denken Sie daran, dass Amazon Managed Service for 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 `kubectl apply` Befehl erneut aus.
> Um eine Ressource zu löschen, führen Sie den folgenden Befehl aus. Ersetzen Sie durch *`ResourceType`* den Ressourcentyp`Workspace`, den Sie löschen möchten `AlertManagerDefinition`, oder `RuleGroupNamespace`. Ersetzen Sie durch *`ResourceName`* 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.

Einen Amazon EKS-Cluster für das Schreiben in den Amazon Managed Service for Prometheus Workspace konfigurieren

In diesem Abschnitt wird beschrieben, wie Sie Helm verwenden, um den in Ihrem Amazon EKS-Cluster ausgeführten Prometheus so zu konfigurieren, dass Metriken durch Remote-Write in den Amazon Managed Service for Prometheus Workspace geschrieben werden, 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 Richten Sie Servicerollen für die Erfassung von Metriken aus Amazon EKS-

Clustern ein. 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. Ersetzen Sie *`WORKSPACE-ID`* durch die Workspace-ID aus dem vorherigen Abschnitt.

   ```
   aws amp describe-workspace --workspace-id WORKSPACE-ID
   ```

   Der Prometheus-Endpunkt wird in den Rückgabeergebnissen enthalten und wie folgt formatiert sein:

   ```
   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`. Ersetzen Sie *`Konto`* durch Ihre Konto-ID, *`workspaceURL/`* durch die URL, die Sie gerade gefunden haben, und *`Region`* durch den passenden AWS-Region für Ihr System.

   ```
   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 Prometheus-Chart- und Namensraum-Namen sowie die Chart-Version mit dem folgenden Helm-Befehl.

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

Basierend auf den bisherigen Schritten sollten das Prometheus-Chart und der Namensraum beide `prometheus` genannt werden, und die Chart-Version kann `15.2.0` genannt werden

4. Führen Sie den folgenden Befehl mit der *PrometheusChartName*, und aus*PrometheusNamespace*, die im vorherigen Schritt *PrometheusChartVersion* gefunden wurden.

```
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 Amazon Managed Service for Prometheus-Endpunkt über `awscurl` abfragen. Ersetzen Sie *Region* durch die , AWS-Region 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 Amazon Managed Service for Prometheus Workspace 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 alle Ihre Amazon Managed Service for Prometheus-Ressourcen direkt vom Cluster aus zu verwalten.

Integrieren von CloudWatch Metriken in Firehose

In diesem Abschnitt wird beschrieben, wie Sie einen Amazon- CloudWatch Metrik-Stream instrumentieren und Amazon Data Firehose und verwendenAWS Lambda, um Metriken in Amazon Managed Service for Prometheus aufzunehmen.

Sie richten einen Stack mit dem AWS Cloud Development Kit (CDK) ein, um einen Firehose-Bereitstellungs-Stream, ein Lambda und einen Amazon 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 Amazon S3-Bucket.

Die Einrichtung der Infrastruktur besteht aus 4 Schritten:

- Voraussetzungen konfigurieren
- Workspace von Amazon Managed Service for Prometheus erstellen
- Installieren von Abhängigkeiten
- Den Stack bereitstellen

Voraussetzungen

- Die AWS CLI wird 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 AWS Github-Repository () des CloudWatch Beobachtbarkeitsmetrik-Exporters wurde auf Ihrem lokalen Computer geklont.`CWMetricsStreamExporter`

So erstellen Sie einen Workspace von Amazon Managed Service for Prometheus

1. Die Demo-App in diesem Rezept wird auf Amazon Managed Service for Prometheus ausgeführt. Erstellen Sie Ihren Amazon Managed Service for Prometheus Workspace 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 for Prometheus finden Sie im Amazon Managed Service for Prometheus Benutzerhandbuch.

So installieren Sie Abhängigkeiten

1. Abhängigkeiten installieren

   Ändern Sie in der Root des `aws-o11y-recipes` Repositorys Ihr Verzeichnis zu `CWMetricStreamExporter` mit dem folgenden Befehl:

   ```
   cd sandbox/CWMetricStreamExporter
   ```

   Dies wird nun in Zukunft als die Root 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 zur Root des Repos und wechseln Sie dann das Verzeichnis zu `/lambda` mit dem folgenden Befehl:

   ```
   cd lambda
   ```

5. Sobald Sie sich im `/lambda` Ordner 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` und ändern Sie in der Root des Repos die Workspace-URL von Amazon Managed Service for Prometheus, indem Sie `{workspace}` durch die neu

erstellte Workspace-ID und die Region ersetzen, in der sich Ihr Amazon Managed Service for Prometheus Workspace 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-Bereitstellungsdatenstroms und des Amazon S3-Buckets nach Ihren Wünschen.

2. Um den AWS CDK und den Lambda-Code zu erstellen, führen Sie im Stammverzeichnis des Repositorys das folgende Empfehlung aus:

```
npm run build
```

Dieser Build-Schritt stellt sicher, dass die Go-Lambda-Binärdatei erstellt wird, und stellt das CDK in 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.

Erstellen eines Amazon- CloudWatch Streams

Nachdem Sie nun über eine Lambda-Funktion zur Verarbeitung der Metriken verfügen, können Sie den Metrik-Stream von Amazon aus erstellen CloudWatch.

So erstellen Sie einen - CloudWatch Metrik-Stream

1. Navigieren Sie zur - CloudWatch Konsole unter https://console.aws.amazon.com/cloudwatch/home#metric-streams:streamsList und wählen Sie Metrik-Stream erstellen aus.

2. Wählen Sie die benötigten Metriken aus, entweder alle Metriken oder nur aus ausgewählten Namensräume.
3. Wählen Sie `Configuration` unter Vorhandenes Firehose auswählen, das Ihrem Konto gehört aus.
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 Metriken werden jetzt von Amazon CloudWatch zu 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. Amazon Managed Service for Prometheus Workspace 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 Metrik-Stream mit der Amazon-CloudWatch Konsole .

Sicherheit in Amazon Managed Service for Prometheus

Die Sicherheit in der Cloud hat für AWS höchste Priorität. Als AWS-Kunde profitieren Sie von Rechenzentren und Netzwerkarchitekturen, die eingerichtet wurden, um die Anforderungen der anspruchsvollsten Organisationen in puncto Sicherheit zu erfüllen.

Sicherheit ist eine übergreifende Verantwortlichkeit zwischen AWS und Ihnen. Das Modell der geteilten Verantwortung beschreibt dies als Sicherheit der Cloud selbst und Sicherheit in der Cloud:

- Sicherheit der Cloud selbst – AWS ist dafür verantwortlich, die Infrastruktur zu schützen, mit der AWS-Services in der AWS Cloud ausgeführt werden. AWS stellt Ihnen außerdem Services bereit, die Sie sicher nutzen können. Auditoren von Drittanbietern testen und überprüfen die Effektivität unserer Sicherheitsmaßnahmen im Rahmen der AWS-Compliance-Programme regelmäßig. Informationen zu den Compliance-Programmen, die für Amazon Managed Service for Prometheus gelten, finden Sie unter Im Rahmen des Compliance-Programms zugelassene AWS-Services.
- Sicherheit in der Cloud – Ihr Verantwortungsumfang wird durch den AWS-Dienst bestimmt, den Sie verwenden. 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 for Prometheus einsetzen können. Die folgenden Themen zeigen Ihnen, wie Sie Amazon Managed Service for Prometheus konfigurieren, um Ihre Sicherheits- und Compliance-Ziele zu erreichen. Sie lernen auch, wie Sie andere AWS-Services verwenden, die Sie bei der Überwachung und Sicherung Ihrer Amazon Managed Service for Prometheus Ressourcen unterstützen.

Themen

- Protokollieren von Amazon Managed Service für Prometheus-API-Aufrufe mithilfe von AWS CloudTrail
- IAM-Rollen für Servicekonten einrichten
- Verwendung von Amazon Managed Service for Prometheus mit Schnittstellen-VPC-Endpunkten

Datenschutz in Amazon Managed Service für Prometheus

Das Modell der AWS gemeinsamen Verantwortung 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 Blog-Beitrag AWS -Modell der geteilten Verantwortung und in der DSGVO im AWS -Sicherheitsblog.

Aus Datenschutzgründen empfehlen wir, dass Sie 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).
- Verwenden Sie SSL/TLS, um mit Ressourcen zu kommunizieren. AWS Wir benötigen TLS 1.2 und empfehlen TLS 1.3.
- Richten Sie die API und die Protokollierung von Benutzeraktivitäten mit ein. AWS CloudTrail
- 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 140-2-validierte kryptografische Module benötigen, verwenden Sie einen FIPS-Endpunkt. Weitere Informationen über verfügbare FIPS-Endpunkte finden Sie unter Federal Information Processing Standard (FIPS) 140-2.

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

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 vom Kunden verwalteten 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 zu Preisen 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, die es Amazon Managed Service for Prometheus ermöglichen, den Schlüssel in Ihrem Namen zu verwenden. 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 Management Console 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 vom Kunden 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 das Auffinden 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

Wie Sie AWS Identity and Access Management (IAM) verwenden, hängt von der Arbeit ab, die Sie in Amazon Managed Service for Prometheus ausführen.

Service-Benutzer – Wenn Sie den Amazon Managed Service für Prometheus zur Ausführung von Aufgaben verwenden, stellt Ihnen Ihr Administrator die Anmeldeinformationen und Berechtigungen bereit, die Sie benötigen. Wenn Sie zur Ausführung von Aufgaben weitere Features von Amazon Managed Service für Prometheus verwenden, benötigen Sie möglicherweise zusätzliche Berechtigungen. Wenn Sie die Fuktionsweise der Zugriffskontrolle nachvollziehen, wissen Sie bereits, welche Berechtigungen Sie von Ihrem Administrator anzufordern müssen. Wenn Sie auf eine Feature in Amazon Managed Service für Prometheus nicht zugreifen können, siehe Problembehebung von Identität und Zugriff auf Amazon Managed Service für Prometheus.

Service-Administrator – Wenn Sie in Ihrem Unternehmen für die Ressourcen von Amazon Managed Service für Prometheus zuständig sind, haben Sie wahrscheinlich vollen Zugriff auf Amazon

Managed Service für Prometheus. Es ist Ihre Aufgabe, zu bestimmen, auf welche Features und Ressourcen von Amazon Managed Service für Prometheus Ihre Service-Benutzer zugreifen sollen. Sie müssen dann Anträge an Ihren IAM-Administrator stellen, um die Berechtigungen Ihrer Servicenutzer zu ändern. Lesen Sie die Informationen auf dieser Seite, um die Grundkonzepte von IAM nachzuvollziehen. Weitere Informationen dazu, wie Ihr Unternehmen IAM mit Amazon Managed Service für Prometheus verwenden kann, finden Sie unter So funktioniert Amazon Managed Service für Prometheus mit IAM.

IAM-Administrator – Als IAM-Administrator möchten Sie vielleicht Details darüber erfahren, wie Sie Richtlinien zur Verwaltung des Zugriffs auf Amazon Managed Service für Prometheus erstellen können. Beispiele für identitätsbasierte Richtlinien von Amazon Managed Service für Prometheus, die Sie in IAM verwenden können, finden Sie unter 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 als IAM-Benutzer authentifiziert (angemeldet AWS) sein oder eine IAM-Rolle annehmen. Root-Benutzer des AWS-Kontos

Sie können sich AWS als föderierte Identität anmelden, indem Sie Anmeldeinformationen verwenden, die über eine Identitätsquelle bereitgestellt wurden. AWS IAM Identity Center (IAM Identity Center) -Benutzer, die Single Sign-On-Authentifizierung Ihres Unternehmens und Ihre Google- oder Facebook-Anmeldeinformationen sind Beispiele für föderierte Identitäten. Wenn Sie sich als Verbundidentität anmelden, hat der Administrator vorher mithilfe von IAM-Rollen einen Identitätsverbund eingerichtet. Wenn Sie über den Verbund darauf zugreifen AWS , übernehmen Sie indirekt eine Rolle.

Je nachdem, welcher Benutzertyp Sie sind, können Sie sich beim AWS Management Console oder beim AWS Zugangsportal anmelden. Weitere Informationen zur Anmeldung finden Sie AWS unter So melden Sie sich bei Ihrem an AWS-Konto im AWS-Anmeldung Benutzerhandbuch.

Wenn Sie AWS programmgesteuert darauf zugreifen, AWS stellt es ein Software Development Kit (SDK) und eine Befehlszeilenschnittstelle (CLI) bereit, um Ihre Anfragen mithilfe Ihrer Anmeldeinformationen kryptografisch zu signieren. Wenn Sie keine AWS Tools verwenden, müssen Sie Anfragen selbst signieren. Weitere Informationen zur Verwendung der empfohlenen Methode, um Anfragen selbst zu signieren, finden Sie im IAM-Benutzerhandbuch unter AWS API-Anfragen signieren.

Unabhängig von der verwendeten Authentifizierungsmethode müssen Sie möglicherweise zusätzliche Sicherheitsinformationen angeben. AWS Empfiehlt beispielsweise, die Multi-Faktor-Authentifizierung (MFA) zu verwenden, um die Sicherheit Ihres Kontos zu erhöhen. Weitere Informationen finden Sie unter Multi-Faktor-Authentifizierung im AWS IAM Identity Center - Benutzerhandbuch und Verwenden der Multi-Faktor-Authentifizierung (MFA) in AWS im IAM-Benutzerhandbuch.

AWS-Konto Root-Benutzer

Wenn Sie einen erstellen AWS-Konto, beginnen Sie mit einer Anmeldeidentität, die vollständigen Zugriff auf alle AWS-Services Ressourcen im Konto hat. Diese Identität wird als AWS-Konto Root-Benutzer bezeichnet. Sie können darauf zugreifen, indem Sie sich mit der E-Mail-Adresse und dem Passwort anmelden, mit denen Sie das Konto erstellt haben. Wir raten ausdrücklich davon ab, den Root-Benutzer für Alltagsaufgaben zu verwenden. Schützen Sie Ihre Root-Benutzer-Anmeldeinformationen und verwenden Sie diese, um die Aufgaben auszuführen, die nur der Root-Benutzer ausführen kann. Eine vollständige Liste der Aufgaben, für die Sie sich als Root-Benutzer anmelden müssen, finden Sie unter Aufgaben, die Root-Benutzer-Anmeldeinformationen erfordern im IAM-Benutzerhandbuch.

Verbundidentität

Als bewährte Methode sollten menschliche Benutzer, einschließlich Benutzer, die Administratorzugriff benötigen, für den Zugriff AWS-Services mithilfe temporärer Anmeldeinformationen den Verbund mit einem Identitätsanbieter verwenden.

Eine föderierte Identität ist ein Benutzer aus Ihrem Unternehmensbenutzerverzeichnis, einem Web-Identitätsanbieter AWS Directory Service, dem Identity Center-Verzeichnis oder einem beliebigen Benutzer, der mithilfe AWS-Services von Anmeldeinformationen zugreift, die über eine Identitätsquelle bereitgestellt wurden. Wenn föderierte Identitäten darauf zugreifen AWS-Konten, übernehmen sie Rollen, und die Rollen stellen temporäre Anmeldeinformationen bereit.

Für die zentrale Zugriffsverwaltung empfehlen wir Ihnen, AWS IAM Identity Center zu verwenden. Sie können Benutzer und Gruppen in IAM Identity Center erstellen, oder Sie können eine Verbindung zu einer Gruppe von Benutzern und Gruppen in Ihrer eigenen Identitätsquelle herstellen und diese synchronisieren, um sie in all Ihren AWS-Konten Anwendungen zu verwenden. Informationen zu IAM Identity Center finden Sie unter Was ist IAM Identity Center? im AWS IAM Identity Center - Benutzerhandbuch.

IAM-Benutzer und -Gruppen

Ein IAM-Benutzer ist eine Identität innerhalb Ihres Unternehmens AWS-Konto , die über spezifische Berechtigungen für eine einzelne Person oder Anwendung verfügt. Wenn möglich, empfehlen wir, temporäre Anmeldeinformationen zu verwenden, anstatt IAM-Benutzer zu erstellen, die langfristige Anmeldeinformationen wie Passwörter und Zugriffsschlüssel haben. Bei speziellen Anwendungsfällen, die langfristige Anmeldeinformationen mit IAM-Benutzern erfordern, empfehlen wir jedoch, die Zugriffsschlüssel zu rotieren. Weitere Informationen finden Sie unter Regelmäßiges Rotieren von Zugriffsschlüsseln für Anwendungsfälle, die langfristige Anmeldeinformationen erfordern im IAM-Benutzerhandbuch.

Eine IAM-Gruppe ist eine Identität, die eine Sammlung von IAM-Benutzern angibt. Sie können sich nicht als Gruppe anmelden. Mithilfe von Gruppen können Sie Berechtigungen für mehrere Benutzer gleichzeitig angeben. Gruppen vereinfachen die Verwaltung von Berechtigungen, wenn es zahlreiche Benutzer gibt. Sie könnten beispielsweise einer Gruppe mit dem Namen IAMAdmins Berechtigungen zum Verwalten von IAM-Ressourcen erteilen.

Benutzer unterscheiden sich von Rollen. Ein Benutzer ist einer einzigen Person oder Anwendung eindeutig zugeordnet. Eine Rolle kann von allen Personen angenommen werden, die sie benötigen. Benutzer besitzen dauerhafte Anmeldeinformationen. Rollen stellen temporäre Anmeldeinformationen bereit. Weitere Informationen finden Sie unter Erstellen eines IAM-Benutzers (anstatt einer Rolle) im IAM-Benutzerhandbuch.

IAM-Rollen

Eine IAM-Rolle ist eine Identität innerhalb Ihres Unternehmens AWS-Konto , die über bestimmte Berechtigungen verfügt. Sie ist einem IAM-Benutzer vergleichbar, ist aber nicht mit einer bestimmten Person verknüpft. Sie können vorübergehend eine IAM-Rolle in der übernehmen, AWS Management Console indem Sie die Rollen wechseln. Sie können eine Rolle übernehmen, indem Sie eine AWS CLI oder AWS API-Operation aufrufen oder eine benutzerdefinierte URL verwenden. Weitere Informationen zu Methoden für die Verwendung von Rollen finden Sie unter Verwenden von IAM-Rollen im IAM-Benutzerhandbuch.

IAM-Rollen mit temporären Anmeldeinformationen sind in folgenden Situationen hilfreich:

- Verbundbenutzerzugriff – Um einer Verbundidentität Berechtigungen zuzuweisen, erstellen Sie eine Rolle und definieren Berechtigungen für die Rolle. Wird eine Verbundidentität authentifiziert, so wird die Identität der Rolle zugeordnet und erhält die von der Rolle definierten Berechtigungen. Informationen zu Rollen für den Verbund finden Sie unter Erstellen von Rollen für externe

Identitätsanbieter im IAM-Benutzerhandbuch. Wenn Sie IAM Identity Center verwenden, konfigurieren Sie einen Berechtigungssatz. Wenn Sie steuern möchten, worauf Ihre Identitäten nach der Authentifizierung zugreifen können, korreliert IAM Identity Center den Berechtigungssatz mit einer Rolle in IAM. Informationen zu Berechtigungssätzen finden Sie unter Berechtigungssätze im AWS IAM Identity Center -Benutzerhandbuch.

- Temporäre IAM-Benutzerberechtigungen – Ein IAM-Benutzer oder eine -Rolle kann eine IAM-Rolle übernehmen, um vorübergehend andere Berechtigungen für eine bestimmte Aufgabe zu erhalten.
- Kontoübergreifender Zugriff – Sie können eine IAM-Rolle verwenden, um einem vertrauenswürdigen Prinzipal in einem anderen Konto den Zugriff auf Ressourcen in Ihrem Konto zu ermöglichen. Rollen stellen die primäre Möglichkeit dar, um kontoübergreifendem Zugriff zu gewähren. Bei einigen können Sie AWS-Services jedoch eine Richtlinie direkt an eine Ressource anhängen (anstatt eine Rolle als Proxy zu verwenden). Informationen zum Unterschied zwischen Rollen und ressourcenbasierten Richtlinien für den kontenübergreifenden Zugriff finden Sie unter Kontenübergreifender Ressourcenzugriff in IAM im IAM-Benutzerhandbuch.
- Serviceübergreifender Zugriff — Einige verwenden Funktionen in anderen. AWS-Services AWS-Services Wenn Sie beispielsweise einen Aufruf in einem Service tätigen, führt dieser Service häufig Anwendungen in Amazon-EC2 aus oder speichert Objekte in Amazon-S3. Ein Dienst kann dies mit den Berechtigungen des aufrufenden Prinzipals mit einer Servicerolle oder mit einer serviceverknüpften Rolle tun.
  - Forward Access Sessions (FAS) — Wenn Sie einen IAM-Benutzer oder eine IAM-Rolle verwenden, um Aktionen auszuführen AWS, gelten Sie als Principal. Bei einigen Services könnte es Aktionen geben, die dann eine andere Aktion in einem anderen Service initiieren. FAS verwendet die Berechtigungen des Prinzipals, der einen aufruft AWS-Service, in Kombination mit der Anfrage, Anfragen an AWS-Service nachgelagerte Dienste zu stellen. FAS-Anfragen werden nur gestellt, wenn ein Dienst eine Anfrage erhält, für deren Abschluss Interaktionen mit anderen AWS-Services oder Ressourcen erforderlich sind. In diesem Fall müssen Sie über Berechtigungen zum Ausführen beider Aktionen verfügen. Einzelheiten zu den Richtlinien für FAS-Anfragen finden Sie unter Zugriffssitzungen weiterleiten.
  - Servicerolle – Eine Servicerolle ist eine IAM-Rolle, die ein Service übernimmt, 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.
  - Dienstbezogene Rolle — Eine dienstbezogene 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. Servicebezogene Rollen erscheinen in Ihrem Dienst AWS-Konto und

gehören dem Dienst. Ein IAM-Administrator kann die Berechtigungen für Service-verknüpfte Rollen anzeigen, aber nicht bearbeiten.

- Auf Amazon EC2 ausgeführte Anwendungen — Sie können eine IAM-Rolle verwenden, um temporäre Anmeldeinformationen für Anwendungen zu verwalten, die auf einer EC2-Instance ausgeführt werden und API-Anfragen stellen AWS CLI . AWS Das ist eher zu empfehlen, als Zugriffsschlüssel innerhalb der EC2-Instance zu speichern. Um einer EC2-Instance eine AWS Rolle zuzuweisen und sie allen ihren Anwendungen zur Verfügung zu stellen, erstellen Sie ein Instance-Profil, das an die Instance angehängt ist. Ein Instance-Profil enthält die Rolle und ermöglicht, dass Programme, die in der EC2-Instance ausgeführt werden, temporäre Anmeldeinformationen erhalten. Weitere Informationen finden Sie unter Verwenden einer IAM-Rolle zum Erteilen von Berechtigungen für Anwendungen, die auf Amazon-EC2-Instances ausgeführt werden im IAM-Benutzerhandbuch.

Informationen dazu, wann Sie IAM-Rollen oder IAM-Benutzer verwenden sollten, finden Sie unter Erstellen einer IAM-Rolle (anstatt eines Benutzers) 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 ist ein Objekt, AWS das, wenn es einer Identität oder Ressource zugeordnet ist, deren Berechtigungen definiert. AWS wertet diese Richtlinien aus, wenn ein Prinzipal (Benutzer, Root-Benutzer oder Rollensitzung) eine Anfrage stellt. Berechtigungen in den Richtlinien bestimmen, ob die Anforderung zugelassen oder abgelehnt wird. Die meisten Richtlinien werden AWS als JSON-Dokumente gespeichert. Weitere Informationen zu Struktur und Inhalten von JSON-Richtliniendokumenten finden Sie unter Übersicht über JSON-Richtlinien im IAM-Benutzerhandbuch.

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer Zugriff auf was hat. Das bedeutet, welcher Prinzipal kann Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen.

Standardmäßig haben Benutzer, Gruppen und Rollen keine Berechtigungen. Ein IAM-Administrator muss IAM-Richtlinien erstellen, die Benutzern die Berechtigung erteilen, Aktionen für die Ressourcen auszuführen, die sie benötigen. Der Administrator kann dann die IAM-Richtlinien zu Rollen hinzufügen, und Benutzer können die Rollen annehmen.

IAM-Richtlinien definieren Berechtigungen für eine Aktion unabhängig von der Methode, die Sie zur Ausführung der Aktion verwenden. Angenommen, es gibt eine Richtlinie, die Berechtigungen für die

`iam:GetRole`-Aktion erteilt. Ein Benutzer mit dieser Richtlinie kann Rolleninformationen von der AWS Management Console AWS CLI, der oder der AWS API abrufen.

Identitätsbasierte Richtlinien

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 Erstellen von IAM-Richtlinien im IAM-Benutzerhandbuch.

Identitätsbasierte Richtlinien können weiter als Inline-Richtlinien oder verwaltete Richtlinien kategorisiert werden. Inline-Richtlinien sind direkt in einen einzelnen Benutzer, eine einzelne Gruppe oder eine einzelne Rolle eingebettet. Verwaltete Richtlinien sind eigenständige Richtlinien, die Sie mehreren Benutzern, Gruppen und Rollen in Ihrem System zuordnen können AWS-Konto. Zu den verwalteten Richtlinien gehören AWS verwaltete Richtlinien und vom Kunden verwaltete Richtlinien. Informationen dazu, wie Sie zwischen einer verwalteten Richtlinie und einer eingebundenen Richtlinie wählen, finden Sie unter Auswahl zwischen verwalteten und eingebundenen Richtlinien im IAM-Benutzerhandbuch.

Ressourcenbasierte Richtlinien

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

Ressourcenbasierte Richtlinien sind Richtlinien innerhalb dieses Diensts. Sie können AWS verwaltete Richtlinien von IAM nicht in einer ressourcenbasierten Richtlinie verwenden.

Zugriffssteuerungslisten (ACLs)

Zugriffssteuerungslisten (ACLs) steuern, welche Prinzipale (Kontomitglieder, Benutzer oder Rollen) auf eine Ressource zugreifen können. ACLs sind ähnlich wie ressourcenbasierte Richtlinien, verwenden jedoch nicht das JSON-Richtliniendokumentformat.

Amazon S3 und Amazon VPC sind Beispiele für Services, die ACLs unterstützen. AWS WAF Weitere Informationen“ zu ACLs finden Sie unter Zugriffskontrollliste (ACL) – Übersicht (Access Control List) im Amazon-Simple-Storage-Service-Entwicklerhandbuch.

Weitere Richtlinientypen

AWS unterstützt zusätzliche, weniger verbreitete Richtlinientypen. Diese Richtlinientypen können die maximalen Berechtigungen festlegen, die Ihnen von den häufiger verwendeten Richtlinientypen erteilt werden können.

- Berechtigungsgrenzen – Eine Berechtigungsgrenze ist ein erweitertes Feature, mit der Sie die maximalen Berechtigungen festlegen können, die eine identitätsbasierte Richtlinie einer IAM-Entität (IAM-Benutzer oder -Rolle) erteilen kann. Sie können eine Berechtigungsgrenze für eine Entität festlegen. Die daraus resultierenden Berechtigungen sind der Schnittpunkt der identitätsbasierten Richtlinien einer Entität und ihrer Berechtigungsgrenzen. Ressourcenbasierte Richtlinien, die den Benutzer oder die Rolle im Feld `Principal` angeben, werden nicht durch Berechtigungsgrenzen eingeschränkt. Eine explizite Zugriffsverweigerung in einer dieser Richtlinien setzt eine Zugriffserlaubnis außer Kraft. Weitere Informationen über Berechtigungsgrenzen finden Sie unter Berechtigungsgrenzen für IAM-Entitäten im IAM-Benutzerhandbuch.
- Service Control Policies (SCPs) — SCPs sind JSON-Richtlinien, die die maximalen Berechtigungen für eine Organisation oder Organisationseinheit (OU) in festlegen. AWS Organizations AWS Organizations ist ein Dienst zur Gruppierung und zentralen Verwaltung mehrerer Objekte AWS-Konten , die Ihrem Unternehmen gehören. Wenn Sie innerhalb einer Organisation alle Features aktivieren, können Sie Service-Kontrollrichtlinien (SCPs) auf alle oder einzelne Ihrer Konten anwenden. Das SCP schränkt die Berechtigungen für Entitäten in Mitgliedskonten ein, einschließlich der einzelnen Entitäten. Root-Benutzer des AWS-Kontos Weitere Informationen zu Organizations und SCPs finden Sie unter Funktionsweise von SCPs im AWS Organizations -Benutzerhandbuch.
- Sitzungsrichtlinien – Sitzungsrichtlinien sind erweiterte Richtlinien, die Sie als Parameter übergeben, wenn Sie eine temporäre Sitzung für eine Rolle oder einen verbundenen Benutzer programmgesteuert erstellen. Die resultierenden Sitzungsberechtigungen sind eine Schnittmenge der auf der Identität des Benutzers oder der Rolle basierenden Richtlinien und der Sitzungsrichtlinien. Berechtigungen können auch aus einer ressourcenbasierten Richtlinie stammen. Eine explizite Zugriffsverweigerung in einer dieser Richtlinien setzt eine Zugriffserlaubnis außer Kraft. Weitere Informationen finden Sie unter Sitzungsrichtlinien im IAM-Benutzerhandbuch.

Mehrere Richtlinientypen

Wenn mehrere auf eine Anforderung mehrere Richtlinientypen angewendet werden können, sind die entsprechenden Berechtigungen komplizierter. Informationen darüber, wie AWS bestimmt wird, ob eine Anfrage zulässig ist, wenn mehrere Richtlinientypen betroffen sind, finden Sie im IAM-Benutzerhandbuch unter Bewertungslogik für Richtlinien.

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	Nein
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 Erstellen von IAM-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. Sie können den Prinzipal nicht in einer identitätsbasierten Richtlinie angeben, da er für den Benutzer oder die Rolle gilt, dem er zugeordnet ist. 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	Nein

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. Durch das Hinzufügen eines kontoübergreifenden Auftraggebers zu einer ressourcenbasierten Richtlinie ist nur die halbe Vertrauensbeziehung eingerichtet. Wenn sich der Prinzipal und die Ressource unterscheiden AWS-Konten, muss ein IAM-Administrator des vertrauenswürdigen Kontos auch der Prinzipalentität (Benutzer oder Rolle) die Berechtigung zum Zugriff auf die Ressource erteilen. Sie erteilen Berechtigungen, indem Sie der juristischen Stelle eine identitätsbasierte Richtlinie anfügen. Wenn jedoch eine ressourcenbasierte Richtlinie Zugriff auf einen Prinzipal in demselben Konto gewährt, ist keine zusätzliche identitätsbasierte Richtlinie erforderlich. Weitere Informationen finden Sie unter Kontenübergreifender 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 auf was Zugriff hat. Das heißt, welcher Prinzipal kann Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen.

Das Element `Action` einer JSON-Richtlinie beschreibt die Aktionen, mit denen Sie den Zugriff in einer Richtlinie zulassen oder verweigern können. Richtlinienaktionen haben normalerweise denselben Namen wie der zugehörige AWS API-Vorgang. Es gibt einige Ausnahmen, z. B. Aktionen, die nur mit Genehmigung durchgeführt werden können und für die es keinen passenden API-Vorgang gibt. Es gibt auch einige Operationen, die mehrere Aktionen in einer Richtlinie erfordern. Diese zusätzlichen Aktionen werden als abhängige Aktionen bezeichnet.

Schließen Sie Aktionen in eine Richtlinie ein, um Berechtigungen zur Durchführung der zugeordneten Operation 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 bedeutet die Festlegung, welcher Prinzipal Aktionen für welche Ressourcen unter welchen Bedingungen ausführen kann.

Das JSON-Richtlinienelement `Resource` gibt die Objekte an, auf welche die Aktion angewendet wird. Anweisungen müssen entweder ein – `Resource`oder ein `NotResource`-Element enthalten. Als bewährte Methode geben Sie eine Ressource mit dem zugehörigen Amazon-Ressourcennamen (ARN) an. Sie können dies für Aktionen tun, die einen bestimmten Ressourcentyp unterstützen, der als Berechtigungen auf Ressourcenebene bezeichnet wird.

Verwenden Sie für Aktionen, die keine Berechtigungen auf Ressourcenebene unterstützen, z. B. Auflistungsoperationen, einen Platzhalter (*), um anzugeben, dass die Anweisung für alle Ressourcen gilt.

```
"Resource": "*"
```

Eine Liste der Ressourcentypen und ihrer ARNs von Amazon Managed Service für Prometheus finden Sie unter Von Amazon Managed Service für Prometheus definierte Ressourcen in der Referenz für die Service-Autorisierung. 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 auf was Zugriff hat. Das heißt, welcher Prinzipal kann Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen.

Das Element `Condition` (oder `Condition` block) ermöglicht Ihnen die Angabe der Bedingungen, unter denen eine Anweisung wirksam ist. Das Element `Condition` ist optional. 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.

Wenn Sie mehrere `Condition`-Elemente in einer Anweisung oder mehrere Schlüssel in einem einzelnen `Condition`-Element angeben, wertet AWS diese mittels einer logischen `AND`-Operation aus. Wenn Sie mehrere Werte für einen einzelnen Bedingungsschlüssel angeben, AWS wertet die Bedingung mithilfe einer logischen `OR` Operation aus. Alle Bedingungen müssen erfüllt werden, bevor die Berechtigungen der Anweisung gewährt werden.

Sie können auch Platzhaltervariablen verwenden, wenn Sie Bedingungen angeben. Beispielsweise können Sie einem IAM-Benutzer die Berechtigung für den Zugriff auf eine Ressource nur dann gewähren, wenn sie mit dessen IAM-Benutzernamen gekennzeichnet ist. Weitere Informationen finden Sie unter IAM-Richtlinienelemente: Variablen und Tags im IAM-Benutzerhandbuch.

AWS unterstützt globale Bedingungsschlüssel und dienstspezifische Bedingungsschlüssel. 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.

Zugriffssteuerungslisten (ACLs) in Amazon Managed Service für Prometheus

Unterstützt ACLs	Nein

Zugriffssteuerungslisten (ACLs) steuern, welche Prinzipale (Kontomitglieder, Benutzer oder Rollen) auf eine Ressource zugreifen können. ACLs sind ähnlich wie ressourcenbasierte 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 definiert werden. In AWS werden diese Attribute als Tags bezeichnet. Sie können Tags an IAM-Entitäten (Benutzer oder Rollen) und an viele AWS Ressourcen anhängen. Das Markieren von Entitäten und Ressourcen ist der erste Schritt von ABAC. Anschließend entwerfen Sie ABAC-Richtlinien, um Operationen zuzulassen, wenn das Tag des Prinzipals mit dem Tag der Ressource übereinstimmt, auf die sie zugreifen möchten.

ABAC ist in Umgebungen hilfreich, die schnell wachsen, und unterstützt Sie in Situationen, in denen die Richtlinienverwaltung mühsam wird.

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 Was ist ABAC? 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

Einige funktionieren AWS-Services nicht, wenn Sie sich mit temporären Anmeldeinformationen anmelden. Weitere Informationen, einschließlich Informationen, die mit temporären Anmeldeinformationen AWS-Services funktionieren AWS-Services , finden Sie im IAM-Benutzerhandbuch unter Diese Option funktioniert mit IAM.

Sie verwenden temporäre Anmeldeinformationen, wenn Sie sich mit einer anderen AWS Management Console Methode als einem Benutzernamen und einem Passwort anmelden. Wenn Sie beispielsweise AWS über den Single Sign-On-Link (SSO) Ihres Unternehmens darauf zugreifen, werden bei diesem Vorgang automatisch temporäre Anmeldeinformationen erstellt. Sie erstellen auch automatisch temporäre Anmeldeinformationen, wenn Sie sich als Benutzer bei der Konsole anmelden und dann die Rollen wechseln. Weitere Informationen zum Wechseln von Rollen finden Sie unter Wechseln zu einer Rolle (Konsole) im IAM-Benutzerhandbuch.

Mithilfe der AWS API AWS CLI oder können Sie temporäre Anmeldeinformationen manuell erstellen. Sie können diese temporären Anmeldeinformationen dann für den Zugriff verwenden AWS. AWS empfiehlt, temporäre Anmeldeinformationen dynamisch zu generieren, anstatt langfristige Zugriffsschlüssel zu verwenden. Weitere Informationen finden Sie unter Temporäre Sicherheitsanmeldeinformationen in IAM.

Forward Access Sessions für Amazon Managed Service for Prometheus

Unterstützt Forward Access Sessions (FAS)	Nein

Wenn Sie einen IAM-Benutzer oder eine IAM-Rolle verwenden, um Aktionen auszuführen AWS, gelten Sie als Principal. Bei einigen Services könnte es Aktionen geben, die dann eine andere Aktion in einem anderen Service initiieren. FAS verwendet die Berechtigungen des Prinzipals, der einen aufruft AWS-Service, kombiniert mit der Anforderung, Anfragen an nachgelagerte Dienste AWS-Service zu stellen. FAS-Anfragen werden nur gestellt, wenn ein Dienst eine Anfrage erhält, für deren Abschluss Interaktionen mit anderen AWS-Services oder Ressourcen erforderlich sind. In diesem Fall müssen Sie über Berechtigungen zum Ausführen beider Aktionen verfügen. 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 dienstbezogene 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. Sie können auch keine Aufgaben mithilfe der AWS API AWS Management Console, AWS Command Line Interface (AWS CLI) oder ausführen. Ein IAM-Administrator muss IAM-Richtlinien erstellen, die Benutzern die Berechtigung

erteilen, Aktionen für die Ressourcen auszuführen, die sie benötigen. Der Administrator kann dann die IAM-Richtlinien zu Rollen hinzufügen, und Benutzer können die Rollen annehmen.

Informationen dazu, wie Sie unter Verwendung dieser beispielhaften JSON-Richtliniendokumente eine identitätsbasierte IAM-Richtlinie erstellen, finden Sie unter Erstellen von IAM-Richtlinien im IAM-Benutzerhandbuch.

Einzelheiten zu Aktionen und Ressourcentypen, die von Amazon Managed Service für Prometheus definiert werden, einschließlich des Formats der ARNs für die einzelnen Ressourcentypen, finden Sie unter Aktionen, Ressourcen und Bedingungsschlüssel für Amazon Managed Service für Prometheus in der Service-Authorisierungs-Referenz.

Themen

Bewährte Methoden 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. Befolgen Sie beim Erstellen oder Bearbeiten identitätsbasierter Richtlinien die folgenden Anleitungen 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 AWS -verwaltete Richtlinien oder AWS -verwaltete Richtlinien für Auftrags-Funktionen 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. AWS 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 zum 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 Konfigurieren eines MFA-geschützten API-Zugriffs im IAM-Benutzerhandbuch.

Weitere Informationen zu bewährten Methoden in IAM finden Sie unter Bewährte Methoden 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 die Erlaubnis, diese Aktion auf der Konsole oder programmgesteuert mithilfe der API oder durchzuführen. 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": "*"
        }
    ]
}
```

AWS verwaltete Richtlinien für Amazon Managed Service für Prometheus

Eine AWS verwaltete Richtlinie ist eine eigenständige Richtlinie, die von erstellt und verwaltet wird. AWS AWS Verwaltete Richtlinien dienen dazu, Berechtigungen für viele gängige Anwendungsfälle bereitzustellen, sodass Sie damit beginnen können, Benutzern, Gruppen und Rollen Berechtigungen zuzuweisen.

Beachten Sie, dass AWS verwaltete Richtlinien für Ihre speziellen Anwendungsfälle möglicherweise keine Berechtigungen mit den geringsten Rechten gewähren, da sie allen AWS Kunden zur Verfügung stehen. Wir empfehlen Ihnen, die Berechtigungen weiter zu reduzieren, indem Sie kundenverwaltete Richtlinien definieren, die speziell auf Ihre Anwendungsfälle zugeschnitten sind.

Sie können die in AWS verwalteten Richtlinien definierten Berechtigungen nicht ändern. Wenn die in einer AWS verwalteten Richtlinie definierten Berechtigungen AWS aktualisiert werden, wirkt sich das Update auf alle Prinzidentitäten (Benutzer, Gruppen und Rollen) aus, denen die Richtlinie zugeordnet ist. AWS aktualisiert eine AWS verwaltete Richtlinie höchstwahrscheinlich, wenn eine neue Richtlinie eingeführt AWS-Service wird oder neue API-Operationen für bestehende Dienste verfügbar werden.

Weitere Informationen finden Sie unter Von AWS verwaltete Richtlinien im IAM-Benutzerhandbuch.

AmazonPrometheusFullAccess

Sie können die `AmazonPrometheusFullAccess`-Richtlinie an Ihre IAM-Identitäten anfügen.

Details zu Berechtigungen

Diese Richtlinie umfasst die folgenden Berechtigungen.

- `aps` – Ermöglicht vollen Zugriff auf Amazon Managed Service für Prometheus

- `eks` – Ermöglicht dem Service Amazon Managed Service für Prometheus, Informationen über Ihre Amazon-EKS-Cluster zu lesen. Dies ist erforderlich, um verwaltete Scraper erstellen und Metriken in Ihrem Cluster ermitteln zu können.
- `ec2` – Ermöglicht es dem Service Amazon Managed Service für Prometheus, Informationen über Ihre Amazon EC2-Netzwerke zu lesen. Dies ist erforderlich, um verwaltete Scraper mit Zugriff auf Ihre Amazon-EKS-Metriken erstellen zu können.
- `iam` – Ermöglicht Prinzipalen das Erstellen einer serviceverknüpften Rolle für Managed Metric Scraper.

Der Inhalt von AmazonPrometheusFullAccesslautet wie folgt:

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Sid": "AllPrometheusActions",
   "Effect": "Allow",
   "Action": [
    "aps:*"
   ],
   "Resource": "*"
  },
  {
   "Sid": "DescribeCluster",
   "Effect": "Allow",
   "Action": [
    "eks:DescribeCluster",
    "ec2:DescribeSubnets",
    "ec2:DescribeSecurityGroups"
   ],
   "Condition": {
    "ForAnyValue:StringEquals": {
     "aws:CalledVia": [
      "aps.amazonaws.com"
     ]
    }
   },
   "Resource": "*"
  },
  {
   "Sid": "CreateServiceLinkedRole",
```

```
   "Effect": "Allow",
   "Action": "iam:CreateServiceLinkedRole",
   "Resource": "arn:aws:iam::*:role/aws-service-role/scraper.aps.amazonaws.com/
AWSServiceRoleForAmazonPrometheusScraper*",
   "Condition": {
    "StringEquals": {
     "iam:AWSServiceName": "scraper.aps.amazonaws.com"
    }
   }
  }
 ]
}
```

AmazonPrometheusConsoleFullAccess

Sie können die `AmazonPrometheusConsoleFullAccess`-Richtlinie an Ihre IAM-Identitäten anfügen.

Details zu Berechtigungen

Diese Richtlinie umfasst die folgenden Berechtigungen.

- `aps` – Ermöglicht vollen Zugriff auf Amazon Managed Service für Prometheus
- `tag` – Ermöglicht Prinzipalen, Tag-Vorschläge in der Konsole von Amazon Managed Service für Prometheus zu sehen.

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Sid": "TagSuggestions",
   "Effect": "Allow",
   "Action": [
    "tag:GetTagValues",
    "tag:GetTagKeys"
   ],
   "Resource": "*"
  },
  {
   "Sid": "PrometheusConsoleActions",
   "Effect": "Allow",
```

```
   "Action": [
    "aps:CreateWorkspace",
    "aps:DescribeWorkspace",
    "aps:UpdateWorkspaceAlias",
    "aps:DeleteWorkspace",
    "aps:ListWorkspaces",
    "aps:DescribeAlertManagerDefinition",
    "aps:DescribeRuleGroupsNamespace",
    "aps:CreateAlertManagerDefinition",
    "aps:CreateRuleGroupsNamespace",
    "aps:DeleteAlertManagerDefinition",
    "aps:DeleteRuleGroupsNamespace",
    "aps:ListRuleGroupsNamespaces",
    "aps:PutAlertManagerDefinition",
    "aps:PutRuleGroupsNamespace",
    "aps:TagResource",
    "aps:UntagResource",
    "aps:CreateLoggingConfiguration",
    "aps:UpdateLoggingConfiguration",
    "aps:DeleteLoggingConfiguration",
    "aps:DescribeLoggingConfiguration"
   ],
   "Resource": "*"
  }
 ]
}
```

AmazonPrometheusRemoteWriteAccess

Der Inhalt von AmazonPrometheusRemoteWriteAccesslautet wie folgt:

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

AmazonPrometheusQueryAccess

Der Inhalt von AmazonPrometheusQueryAccesslautet wie folgt:

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

AWS verwaltete Richtlinie: AmazonPrometheusScraperServiceRolePolicy

Sie können keine Verbindungen AmazonPrometheusScraperServiceRolePolicy zu Ihren IAM-Entitäten herstellen. Diese Richtlinie ist einer serviceverknüpften Rolle zugeordnet, die Amazon Managed Service für Prometheus erlaubt, Aktionen in Ihrem Namen durchzuführen. Weitere Informationen finden Sie unter Verwenden von Rollen zum Abrufen von Metriken aus EKS.

Diese Richtlinie gewährt Mitwirkenden Berechtigungen, die es ermöglichen, aus Ihrem Amazon-EKS-Cluster zu lesen und in Ihren Workspace in Amazon Managed Service für Prometheus zu schreiben.

> **Note**
>
> In diesem Benutzerhandbuch wurde diese Richtlinie zuvor fälschlicherweise als Richtlinie bezeichnet `AmazonPrometheusScraperServiceLinkedRolePolicy`

Details zu Berechtigungen

Diese Richtlinie umfasst die folgenden Berechtigungen.

- `aps` – Ermöglicht dem Service-Prinzipal, Metriken in Ihre Workspaces von Amazon Managed Service für Prometheus zu schreiben.

- `ec2` – Ermöglicht dem Service-Prinzipal, die Netzwerkkonfiguration zu lesen und zu ändern, um eine Verbindung mit dem Netzwerk herzustellen, das Ihre Amazon-EKS-Cluster enthält.
- `eks` – Ermöglicht dem Service-Prinzipal den Zugriff auf Ihre Amazon-EKS-Cluster. Dies ist erforderlich, damit Metriken automatisch erfasst werden können. Ermöglicht dem Principal außerdem, Amazon EKS-Ressourcen zu bereinigen, wenn ein Scraper entfernt wird.

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Sid": "DeleteSLR",
   "Effect": "Allow",
   "Action": [
    "iam:DeleteRole"
   ],
   "Resource": "arn:aws:iam::*:role/aws-service-role/scraper.aps.amazonaws.com/
AWSServiceRoleForAmazonPrometheusScraper*"
  },
  {
   "Sid": "NetworkDiscovery",
   "Effect": "Allow",
   "Action": [
    "ec2:DescribeNetworkInterfaces",
    "ec2:DescribeSubnets",
    "ec2:DescribeSecurityGroups"
   ],
   "Resource": "*"
  },
  {
   "Sid": "ENIManagement",
   "Effect": "Allow",
   "Action": "ec2:CreateNetworkInterface",
   "Resource": "*",
   "Condition": {
    "ForAllValues:StringEquals": {
     "aws:TagKeys": [
      "AMPAgentlessScraper"
     ]
    }
   }
  },
```

```
{
 "Sid": "TagManagement",
 "Effect": "Allow",
 "Action": "ec2:CreateTags",
 "Resource": "arn:aws:ec2:*:*:network-interface/*",
 "Condition": {
  "StringEquals": {
   "ec2:CreateAction": "CreateNetworkInterface"
  },
  "Null": {
   "aws:RequestTag/AMPAgentlessScraper": "false"
  }
 }
},
{
 "Sid": "ENIUpdating",
 "Effect": "Allow",
 "Action": [
  "ec2:DeleteNetworkInterface",
  "ec2:ModifyNetworkInterfaceAttribute"
 ],
 "Resource": "*",
 "Condition": {
  "Null": {
   "ec2:ResourceTag/AMPAgentlessScraper": "false"
  }
 }
},
{
 "Sid": "EKSAccess",
 "Effect": "Allow",
 "Action": "eks:DescribeCluster",
 "Resource": "arn:aws:eks:*:*:cluster/*"
},
{
 "Sid": "DeleteEKSAccessEntry",
 "Effect": "Allow",
 "Action": "eks:DeleteAccessEntry",
 "Resource": "arn:aws:eks:*:*:access-entry/*/role/*",
 "Condition": {
  "StringEquals": {
   "aws:PrincipalAccount": "${aws:ResourceAccount}"
  },
  "ArnLike": {
```

```
     "eks:principalArn": "arn:aws:iam::*:role/aws-service-role/
scraper.aps.amazonaws.com/AWSServiceRoleForAmazonPrometheusScraper*"
    }
   }
  },
  {
   "Sid": "APSWriting",
   "Effect": "Allow",
   "Action": "aps:RemoteWrite",
   "Resource": "arn:aws:aps:*:*:workspace/*",
   "Condition": {
    "StringEquals": {
     "aws:PrincipalAccount": "${aws:ResourceAccount}"
    }
   }
  }
 ]
}
```

Amazon Managed Service für Prometheus aktualisiert verwaltete Richtlinien AWS

Sehen Sie sich Details zu Aktualisierungen der AWS verwalteten Richtlinien für Amazon Managed Service for Prometheus an, seit dieser Service begonnen hat, diese Änderungen zu verfolgen. Um automatische Benachrichtigungen über Änderungen an dieser Seite zu erhalten, abonnieren Sie den RSS-Feed auf der Dokumentverlauf-Seite von Amazon Managed Service für Prometheus.

Änderung	Beschreibung	Datum
AmazonPrometheusScraperServiceRolePolicy – Aktualisierung auf eine bestehende Richtlinie	Amazon Managed Service for Prometheus hat neue Berechtigungen hinzugefügt, AmazonPrometheusScraperServiceRolePolicyum die Verwendung von Zugriffseinträgen in Amazon EKS zu unterstützen. Beinhaltet Berechtigungen für die Verwaltung von Amazon EKS-Zugriffseinträgen, um	2. Mai 2024

Änderung	Beschreibung	Datum
	Ressourcen zu bereinige n, wenn Scraper gelöscht werden. **Note** In der Bedienung sanleitung wurde diese Richtlinie zuvor fälschlicherweise genannt `AmazonPro metheusSc raperServ iceLinked RolePolicy`	
AmazonPrometheusFu llAccess – Aktualisierung auf eine bestehende Richtlinie	Amazon Managed Service für Prometheus hat AmazonPro metheusFullAccess neue Berechtigungen hinzugefügt, um die Erstellung verwaltet er Scraper für Metriken in Amazon-EKS-Clustern zu unterstützen. Beinhaltet Berechtigungen zum Herstellen einer Verbindung zu Amazon-EK S-Clustern, zum Lesen von Amazon-EC2-Netzwerken und zum Erstellen einer serviceve rknüpften Rolle für Scraper.	26. November 2023

Änderung	Beschreibung	Datum
AmazonPrometheusScraperServiceLinkedRolePolicy – Neue Richtlinie.	Amazon Managed Service für Prometheus hat eine neue servicebezogene Rollenrichtlinie hinzugefügt, die aus Amazon-EKS-Containern gelesen werden kann, um das automatische Scraping von Metriken zu ermöglichen. Beinhaltet Berechtigungen zum Herstellen einer Verbindung mit Amazon-EKS-Clustern, zum Lesen von Amazon-EC2-Netzwerken und zum Erstellen und Löschen von Netzwerken, die als `AMPAgentlessScraper` markiert sind, sowie zum Schreiben in den Workspaces von Amazon Managed Service für Prometheus.	26. November 2023

Änderung	Beschreibung	Datum
AmazonPrometheusConsoleFullAccess – Aktualisierung auf eine bestehende Richtlinie	Amazon Managed Service for Prometheus hat neue Berechtigungen hinzugefügt, AmazonPrometheusConsoleFullAccessum die Protokollierung von Alert Manager- und Lineal-Ereignissen in CloudWatch Logs zu unterstützen. Die`aps:CreateLoggingConfiguration` ,`aps:UpdateLoggingConfiguration` , `aps:DeleteLoggingConfiguration` , `aps:DescribeLoggingConfiguration` Berechtigungen wurden hinzugefügt.	24. Oktober 2022

Änderung	Beschreibung	Datum
AmazonPrometheusConsoleFullAccess – Aktualisierung auf eine bestehende Richtlinie	Amazon Managed Service für Prometheus hat AmazonPrometheusConsoleFullAccess neue Berechtigungen hinzugefügt, um neue Funktionen von Amazon Managed Service für Prometheus zu unterstützen, sodass Benutzer mit dieser Richtlinie eine Liste mit Tag-Vorschlägen sehen können, wenn sie Tags auf die Ressourcen für Amazon Managed Service für Prometheus anwenden. Die `tag:GetTagKeys`, `tag:GetTagValues`, `aps:CreateAlertManagerDefinition`, `aps:CreateRuleGroupsNamespace`, `aps:DeleteAlertManagerDefinition`, `aps:DeleteRuleGroupsNamespace`, `aps:DescribeAlertManagerDefinition`, `aps:DescribeRuleGroupsNamespace`, `aps:ListRuleGroupsNamespaces`, `aps:PutAlertManagerDefinition`, `aps:PutRuleGroupsNamespace`,	29. September 2021

Änderung	Beschreibung	Datum
	`aps:TagResource` und `aps:UntagResource` Berechtigungen wurden hinzugefügt.	
Amazon Managed Service für Prometheus hat mit der Verfolgung von Änderungen begonnen	Amazon Managed Service for Prometheus hat damit begonnen, Änderungen an seinen AWS verwalteten Richtlinien nachzuverfolgen.	15. September 2021

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 durchzuführen: 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 durchzuführen: 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, die Benutzer in anderen Konten oder Personen außerhalb Ihrer Organisation für den Zugriff auf Ihre Ressourcen verwenden können. Sie können festlegen, wem

die Übernahme der Rolle anvertraut wird. Im Fall von Diensten, die ressourcenbasierte Richtlinien oder Zugriffskontrolllisten (Access Control Lists, ACLs) verwenden, 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 gewähren können, AWS-Konten die Ihnen gehören, finden Sie im IAM-Benutzerhandbuch unter Gewähren des Zugriffs auf einen IAM-Benutzer in einem anderen AWS-Konto , den Sie besitzen.
- 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 im IAM-Benutzerhandbuch unter Kontenübergreifender Ressourcenzugriff in IAM.

IAM-Berechtigungen und Richtlinien

Für den Zugriff auf Aktionen und Daten von Amazon Managed Service for Prometheus sind Anmeldeinformationen erforderlich. Diese Anmeldeinformationen müssen über Berechtigungen für die Ausführung der Aktionen und für den Zugriff auf die AWS Ressourcen wie das Abrufen von Amazon Managed Service for Prometheus Daten über Ihre Cloud-Ressourcen verfügen. In den folgenden Abschnitten erfahren Sie, wie Sie Ihre Ressourcen mithilfe von AWS Identity and Access Management (IAM) und Amazon Managed Service for Prometheus sichern können, indem Sie den Zugriff darauf kontrollieren: Weitere Informationen finden Sie unter Richtlinien und Berechtigungen in IAM.

Berechtigungen für Amazon Managed Service für Prometheus

In der folgenden Tabelle werden mögliche Aktionen von Amazon Managed Service for Prometheus und die erforderlichen Berechtigungen aufgeführt. Für die Aktionen sind möglicherweise auch Berechtigungen von anderen Services erforderlich, auf die hier nicht näher eingegangen wird.

Action	Erforderliche Berechtigung
Erstellen Sie Alerts.	`aps:CreateAlertManagerAlerts`
Erstellen Sie eine Alert Manager-Definition in einem Workspace. Weitere Informationen finden Sie unter Alert Manager.	`aps:CreateAlertManagerDefinition`
Erstellen Sie einen Regelgruppen-Namensraum in einem Workspace. Weitere Informationen finden Sie unter Aufzeichnungs- und Alarmregeln.	`aps:CreateRuleGroupsNamespace`
Erstellen Sie einen Amazon Managed Service für Prometheus Workspace. Ein Workspace ist ein logischer Bereich, der der Speicherung und Abfrage von Prometheus-Metriken gewidmet ist.	`aps:CreateWorkspace`
Löschen einer Alert Manager-Definition aus einem Workspace.	`aps:DeleteAlertManagerDefinition`
Löschen Sie stumme Alerts.	`aps:DeleteAlertManagerSilence`
Löschen Sie einen Amazon Managed Service for Prometheus Workspace.	`aps:DeleteWorkspace`
Rufen Sie detaillierte Informationen zu Alert Manager-Definitionen ab.	`aps:DescribeAlertManagerDefinition`
Rufen Sie detaillierte Informationen zu den Namensräumen für Regelgruppen ab.	`aps:DescribeRuleGroupsNamespace`
Rufen Sie detaillierte Informationen über einen Amazon Managed Service for Prometheus Workspace ab.	`aps:DescribeWorkspace`

Action	Erforderliche Berechtigung
Rufen Sie detaillierte Informationen über einen stummen Alert ab.	`aps:GetAlertManagerSilence`
Rufen Sie den Status des Alert Managers in einem Workspace ab.	`aps:GetAlertManagerStatus`
Rufen Sie Etiketten ab.	`aps:GetLabels`
Rufen Sie Metadaten für Amazon Managed Service for Prometheus-Metriken ab.	`aps:GetMetricMetadata`
Rufen Sie Zeitreihendaten ab.	`aps:GetSeries`
Rufen Sie eine Liste der Alert-Gruppen ab, die in der Alert Manager-Definition definiert sind.	`aps:ListAlertManagerAlertGroups`
Rufen Sie eine Liste der Alerts ab, die im Alert Manager definiert sind.	`aps:ListAlertManagerAlerts`
Rufen Sie eine Liste der Empfänger ab, die in der Alert Manager-Definition definiert sind.	`aps:ListAlertManagerReceivers`
Rufen Sie eine Liste der definierten stummen Alerts ab.	`aps:ListAlertManagerSilences`
Rufen Sie eine Liste der aktiven Alerts ab.	`aps:ListAlerts`
Rufen Sie eine Liste der Regeln in den Regelgruppen-Namensräumen in Ihren Workspaces ab.	`aps:ListRules`
Rufen Sie eine Liste der Regelgruppen-Namensräumen in Ihren Workspaces ab.	`aps:ListRuleGroupsNamespaces`

Action	Erforderliche Berechtigung
Rufen Sie die Tags ab, die mit Ihren Amazon Managed Service for Prometheus-Ressourcen verknüpft sind.	`aps:ListTagsForResource`
Rufen Sie eine Liste der Amazon Managed Service for Prometheus Workspaces ab, die im Konto vorhanden sind.	`aps:ListWorkspaces`
Aktualisieren Sie eine bestehende Alert Manager-Definition in einem Workspace.	`aps:PutAlertManagerDefinition`
Erstellen Sie Stummschaltungen bei Alerts.	`aps:PutAlertManagerSilences`
Aktualisieren Sie einen vorhandenen Regelgruppen-Namensraum.	`aps:PutRuleGroupsNamespace`
Führen Sie eine Abfrage von Amazon Managed Service for Prometheus-Metriken aus.	`aps:QueryMetrics`
Führen Sie eine Remote-Write-Operation durch, um das Streaming von Metriken von einem Prometheus-Server an Amazon Managed Service for Prometheus zu initiieren.	`aps:RemoteWrite`
Weisen Sie Amazon Managed Service for Prometheus-Ressourcen Tags zu	`aps:TagResource`
Entfernen Sie Tags aus Amazon Managed Service for Prometheus-Ressourcen.	`aps:UntagResource`
Ändern Sie die Aliase vorhandener Workspaces.	`aps:UpdateWorkspaceAlias`

Action	Erforderliche Berechtigung
Erstellen Sie eine Konfiguration für die Protokollierung.	`aps:CreateLoggingConfiguration`
Löschen Sie eine Konfiguration für die Protokollierung.	`aps:DeleteLoggingConfiguration`
Beschreiben Sie die Konfiguration der Workspace-Protokollierung.	`aps:DescribeLoggingConfiguration`
Aktualisieren Sie eine Protokollierungskonfiguration.	`aps:UpdateLoggingConfiguration`

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 for Prometheus und ermöglicht es einem Benutzer außerdem, Amazon EKS-Cluster zu entdecken und deren Details einzusehen.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "aps:*",
                "eks:DescribeCluster",
                "eks:ListClusters"
            ],
            "Resource": "*"
        }
    ]
}
```

Compliance-Validierung für Amazon Managed Service für Prometheus

Informationen darüber, ob AWS-Service ein AWS-Services in den Geltungsbereich bestimmter Compliance-Programme fällt, finden Sie unter Umfang nach Compliance-Programm AWS-Services unter . Wählen Sie dort das Compliance-Programm aus, an dem Sie interessiert sind. Allgemeine Informationen finden Sie unter AWS Compliance-Programme AWS .

Sie können Prüfberichte von Drittanbietern unter herunterladen AWS Artifact. Weitere Informationen finden Sie unter Berichte herunterladen unter .

Ihre Verantwortung für die Einhaltung der Vorschriften bei der Nutzung AWS-Services hängt von der Vertraulichkeit Ihrer Daten, den Compliance-Zielen Ihres Unternehmens und den geltenden Gesetzen und Vorschriften ab. AWS stellt die folgenden Ressourcen zur Verfügung, die Sie bei der Einhaltung der Vorschriften unterstützen:

- Schnellstartanleitungen zu Sicherheit und Compliance — In diesen Bereitstellungsleitfäden werden architektonische Überlegungen erörtert und Schritte für die Implementierung von Basisumgebungen beschrieben AWS , bei denen Sicherheit und Compliance im Mittelpunkt stehen.
- Architecting for HIPAA Security and Compliance on Amazon Web Services — In diesem Whitepaper wird beschrieben, wie Unternehmen HIPAA-fähige Anwendungen erstellen AWS können.

  > **Note**
  >
  > AWS-Services Nicht alle sind HIPAA-fähig. Weitere Informationen finden Sie in der Referenz für HIPAA-berechtigte Services.

- AWS Compliance-Ressourcen — Diese Sammlung von Arbeitsmappen und Leitfäden gilt möglicherweise für Ihre Branche und Ihren Standort.
- AWS Leitfäden zur Einhaltung von Vorschriften für Kunden — Verstehen Sie das Modell der gemeinsamen Verantwortung aus dem Blickwinkel der Einhaltung von Vorschriften. In den Leitfäden werden die bewährten Verfahren zur Sicherung zusammengefasst AWS-Services und die Leitlinien den Sicherheitskontrollen in verschiedenen Frameworks (einschließlich des National Institute of Standards and Technology (NIST), des Payment Card Industry Security Standards Council (PCI) und der International Organization for Standardization (ISO)) zugeordnet.

- Evaluierung von Ressourcen anhand von Regeln im AWS Config Entwicklerhandbuch — Der AWS Config Service bewertet, wie gut Ihre Ressourcenkonfigurationen den internen Praktiken, Branchenrichtlinien und Vorschriften entsprechen.
- AWS Security Hub— Auf diese AWS-Service Weise erhalten Sie einen umfassenden Überblick über Ihren internen Sicherheitsstatus. AWS Security Hub verwendet Sicherheitskontrollen, um Ihre AWS -Ressourcen zu bewerten und Ihre Einhaltung von Sicherheitsstandards und bewährten Methoden zu überprüfen. Eine Liste der unterstützten Services und Kontrollen finden Sie in der Security-Hub-Steuerungsreferenz.
- Amazon GuardDuty — Dies AWS-Service erkennt potenzielle Bedrohungen für Ihre Workloads AWS-Konten, Container und Daten, indem es Ihre Umgebung auf verdächtige und böswillige Aktivitäten überwacht. GuardDuty kann Ihnen helfen, verschiedene Compliance-Anforderungen wie PCI DSS zu erfüllen, indem es die in bestimmten Compliance-Frameworks vorgeschriebenen Anforderungen zur Erkennung von Eindringlingen erfüllt.
- AWS Audit Manager— Auf diese AWS-Service Weise können Sie Ihre AWS Nutzung kontinuierlich überprüfen, um das Risikomanagement und die Einhaltung von Vorschriften und Industriestandards zu vereinfachen.

Ausfallsicherheit in Amazon Managed Service for Prometheus

Im Zentrum der globalen AWS Infrastruktur stehen die AWS-Regionen und Availability Zones (Verfügbarkeitszonen, AZs). AWS -Regionen stellen mehrere physisch getrennte und isolierte Availability Zones bereit, die über hoch redundante Netzwerke mit niedriger Latenz und hohen Durchsätzen 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 über AWS-Regionen und -Availability Zones finden Sie unter AWS Globale Infrastruktur.

Neben der AWS globalen Infrastruktur stellt Amazon Managed Service for Prometheus verschiedene Features bereit, um Ihren Anforderungen an Ausfallsicherheit und Datensicherung gerecht zu werden, darunter Unterstützung für Daten mit hoher Verfügbarkeit.

Sicherheit der Infrastruktur in Amazon Managed Service for Prometheus

Als verwalteter Service ist Amazon Managed Service for Prometheus durch die globale Netzwerksicherheit von AWS geschützt. Informationen zu AWS-Sicherheitsservices und wie AWS die Infrastruktur schützt, finden Sie unter AWS-Cloud-Sicherheit. Informationen zum Entwerfen Ihrer AWS-Umgebung anhand der bewährten Methoden für die Infrastruktursicherheit finden Sie unter Infrastrukturschutz im Security Pillar AWS Well-Architected Framework.

Sie verwenden durch 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.

Außerdem müssen Anforderungen mit einer Zugriffsschlüssel-ID und einem geheimen Zugriffsschlüssel signiert sein, der einem IAM-Prinzipal zugeordnet ist. Alternativ können Sie mit AWS Security Token Service (AWS STS) temporäre Sicherheitsanmeldeinformationen erstellen, um die Anforderungen zu signieren.

Verwenden von serviceverknüpften Rollen für Amazon Managed Service für Prometheus

Amazon Managed Service for Prometheus verwendet AWS Identity and Access Management (IAM) service-verknüpfte Rollen. Eine serviceverknüpfte Rolle ist ein spezieller Typ einer IAM-Rolle, die direkt mit Amazon Managed Service für Prometheus verknüpft ist. Serviceverknüpfte Rollen werden von Amazon Managed Service für Prometheus vordefiniert und schließen alle Berechtigungen ein, die der Service zum Aufrufen anderer AWS -Services in Ihrem Namen benötigt.

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 die AWSServiceRoleForAmazonPrometheusScraper serviceverknüpfte Rolle 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 Services, die serviceorientierte Rollen unterstützen, finden Sie unter AWS services that work with IAM (-Services, die mit IAM funktionieren). Suchen Sie nach den Services, für die Yes (Ja) in der Spalte Service-linked roles (Serviceorientierte Rollen) angegeben ist. Wählen Sie über einen Link Ja aus, um die Dokumentation zu einer serviceverknüpften Rolle für diesen Service anzuzeigen.

Serviceverknüpfte Rollenberechtigungen für Amazon Managed Service für Prometheus

Amazon Managed Service for Prometheus verwendet eine serviceverknüpfte Rolle, die mit dem Präfix AWSServiceRoleForAmazonPrometheusScraperbenannt ist, damit Amazon Managed Service for Prometheus automatisch Metriken in Ihren Amazon EKS-Clustern auslesen kann.

Die AWSServiceRoleForAmazonPrometheusScraper servicebezogene Rolle vertraut darauf, dass die folgenden Dienste die Rolle übernehmen:

- `scraper.aps.amazonaws.com`

Die genannte Rollenberechtigungsrichtlinie AmazonPrometheusScraperServiceRolePolicyermöglicht es Amazon Managed Service for Prometheus, die folgenden Aktionen für die angegebenen Ressourcen durchzuführen:

- Bereiten Sie die Netzwerkkonfiguration vor und ändern Sie sie, 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.

Sie müssen Berechtigungen konfigurieren, damit Ihre Benutzer, Gruppen oder Rollen eine serviceverknüpfte Rolle erstellen können. Weitere Informationen finden Sie unter serviceverknüpfte Rollenberechtigung im 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 Management Console, der oder der AWS API erstellen AWS CLI, erstellt Amazon Managed Service for Prometheus 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 ist erschienen 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 for Prometheus erlaubt Ihnen nicht, die AWSServiceRoleForAmazonPrometheusScraper serviceverknüpfte Rolle zu bearbeiten. 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 die Rolle nicht manuell löschen. AWSServiceRoleForAmazonPrometheusScraper Wenn Sie alle verwalteten Collector-Instances löschen, die der Rolle in der AWS Management

Console AWS CLI, der oder der AWS API zugeordnet sind, bereinigt Amazon Managed Service for Prometheus 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 für Prometheus-API-Aufrufe mithilfe von AWS CloudTrail

Amazon Managed Service for Prometheus ist in integriert AWS CloudTrail. Dieser Service zeichnet die Aktionen eines Benutzers, einer Rolle oder eines - AWS Services in Amazon Managed Service for Prometheus auf. CloudTrail erfasst alle API-Aufrufe für Amazon Managed Service for Prometheus als Ereignisse. Zu den Aufrufen, die erfasst werden, gehören Aufrufe von der Amazon Managed Service for Prometheus Konsole und Codeaufrufe der Amazon Managed Service for Prometheus API-Operationen. Wenn Sie einen Trail erstellen, können Sie die kontinuierliche Bereitstellung von CloudTrail Ereignissen an einen Amazon S3-Bucket aktivieren, einschließlich Ereignissen für Amazon Managed Service for Prometheus. Wenn Sie keinen Trail konfigurieren, können Sie trotzdem die neuesten Ereignisse in der CloudTrail Konsole unter Ereignisverlauf anzeigen. Anhand der von CloudTrailgesammelten Informationen können Sie die an Amazon Managed Service for Prometheus gestellte Anfrage, die IP-Adresse, von der die Anfrage gestellt wurde, den Initiator der Anfrage, den Zeitpunkt der Anfrage und zusätzliche Details bestimmen.

Weitere Informationen zu CloudTrailfinden Sie im AWS CloudTrail -Benutzerhandbuch.

Informationen zu Amazon Managed Service for Prometheus in CloudTrail

CloudTrail wird beim Erstellen des AWS Kontos in Ihrem Konto aktiviert. Wenn eine Aktivität in Amazon Managed Service for Prometheus auftritt, wird diese Aktivität in einem - CloudTrail Ereignis zusammen mit anderen AWS -Serviceereignissen im Ereignisverlauf aufgezeichnet. Sie können aktuelle Ereignisse in Ihrem AWS Konto anzeigen, suchen und herunterladen. Weitere Informationen finden Sie unter Anzeigen von Ereignissen mit dem CloudTrail Ereignisverlauf .

Erstellen Sie für eine fortlaufende Aufzeichnung der Ereignisse in Ihrem AWS Konto, einschließlich Ereignissen für Amazon Managed Service for Prometheus, einen Trail. Ein Trail ermöglicht CloudTrail

die Bereitstellung von Protokolldateien an einen Amazon S3-Bucket. Wenn Sie einen Trail in der Konsole erstellen, gilt der Trail standardmäßig für alle AWS Regionen. Der Trail protokolliert Ereignisse aus allen Regionen in der AWS -Partition und stellt die Protokolldateien in dem von Ihnen angegebenen Amazon S3-Bucket bereit. Darüber hinaus können Sie andere - AWS Services konfigurieren, um die in den CloudTrail Protokollen erfassten Ereignisdaten weiter zu analysieren und entsprechend zu agieren. Weitere Informationen finden Sie hier:

- Übersicht zum Erstellen eines Trails
- CloudTrail Von unterstützte Services und Integrationen
- Konfigurieren von Amazon SNS-Benachrichtigungen für CloudTrail
- Empfangen von CloudTrail Protokolldateien aus mehreren Regionen und Empfangen von CloudTrail Protokolldateien aus mehreren Konten

Amazon Managed Service for Prometheus unterstützt die Protokollierung der folgenden Aktionen:

- CreateAlertManagerAlerts
- CreateAlertManagerDefinition
- CreateRuleGroupsNamespace
- CreateWorkspace
- DeleteAlertManagerDefinition
- DeleteAlertManagerSilence
- DeleteWorkspace
- DeleteRuleGroupsNamespace
- DescribeAlertManagerDefinition
- DescribeRulesGroupsNamespace
- DescribeWorkspace
- ListRuleGroupsNamespaces
- ListWorkspaces
- PutAlertManagerDefinition
- PutAlertManagerSilences
- PutRuleGroupsNamespace
- UpdateWorkspaceAlias

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 Root- oder AWS Identity and Access Management (IAM)-Benutzeranmeldeinformationen ausgeführt wurde.
- Gibt an, ob die Anforderung mit temporären Sicherheitsanmeldeinformationen für eine Rolle oder einen Verbundbenutzer gesendet wurde.
- Ob die Anforderung von einem anderen - AWS Service gestellt wurde.

Weitere Informationen finden Sie unter CloudTrail -Element userIdentity.

zu den Protokolldateieinträgen von Amazon Managed Service for Prometheus verstehen

Ein Trail ist eine Konfiguration, die die Bereitstellung von Ereignissen als Protokolldateien an einen von Ihnen angegebenen Amazon S3-Bucket ermöglicht. CloudTrail Protokolldateien enthalten einen oder mehrere Protokolleinträge. Ein Ereignis stellt eine einzelne Anforderung aus einer beliebigen Quelle dar und enthält Informationen über die angeforderte Aktion, das Datum und die Uhrzeit der Aktion, Anforderungsparameter usw. CloudTrail Protokolldateien sind kein geordnetes Stacktrace der öffentlichen API-Aufrufe und erscheinen daher nicht in einer bestimmten Reihenfolge.

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"
}
```


IAM-Rollen für Servicekonten einrichten

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.

IAM-Rollen für Servicekonten werden auch als Servicerollen bezeichnet.

In Amazon Managed Service for Prometheus können Sie mithilfe von Servicerollen die Rollen erhalten, die Sie für die Autorisierung und Authentifizierung zwischen Amazon Managed Service for Prometheus, Prometheus-Servern und Grafana-Servern benötigen.

Voraussetzungen

Für die Verfahren auf dieser Seite müssen Sie die Befehlszeilenschnittstelle AWS CLI und EKSCTL installiert haben.

Richten Sie Servicerollen für die Erfassung von Metriken aus Amazon EKS-Clustern ein

Um die Servicerollen so einzurichten, dass Amazon Managed Service for Prometheus Metriken von Prometheus-Servern in Amazon EKS-Clustern aufnehmen 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 Aufnahme in Amazon Managed Service for 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-Namensraum.

```
#!/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

Um die IAM-Rolle für ein Servicekonto (Servicerolle) einzurichten, um die Abfrage von Metriken aus Amazon Managed Service for Prometheus Workspaces zu ermöglichen, müssen Sie bei einem Konto mit den folgenden Berechtigungen angemeldet sein:

- `iam:CreateRole`
- `iam:CreatePolicy`
- `iam:GetRole`
- `iam:AttachRolePolicy`
- `iam:GetOpenIDConnectProvider`

Um Servicerollen für die Abfrage von Amazon Managed Service for Prometheus-Metriken 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>ersetzen Sie es durch Ihren Prometheus-Namensraum.

```
#!/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 for Prometheus mit Schnittstellen-VPC-Endpunkten

Wenn Sie Amazon Virtual Private Cloud (Amazon VPC) zum Hosten Ihrer AWS-Ressourcen verwenden, können Sie private Verbindungen zwischen Ihrem VPC und Amazon Managed Service for Prometheus herstellen. Sie können diese Verbindungen verwenden, damit Amazon Managed Service for Prometheus mit den Ressourcen in der VPC kommunizieren kann, ohne das öffentliche Internet verwenden zu müssen.

Amazon VPC ist ein AWS-Service, den Sie verwenden können, um AWS-Ressourcen in einem von Ihnen definierten virtuellen Netzwerk auszuführen. 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 for Prometheus zu verbinden, definieren Sie einen Schnittstellen-VPC-Endpunkt, um Ihre VPC mit AWSServices zu verbinden. Der Endpunkt bietet zuverlässige, skalierbare Konnektivität zu Amazon Managed Service for 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 über AWS PrivateLink bereitgestellt, eine AWS-Technologie, die eine private Kommunikation zwischen AWS-Services unter Verwendung einer Elastic Network-Schnittstelle mit privaten IP-Adressen ermöglicht. Weitere Informationen finden Sie im Blogbeitrag Neu – AWS PrivateLink für AWS-Services.

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 for Prometheus

Erstellen eines Schnittstellen-VPC-Endpunkts, um mit der Nutzung von Amazon Managed Service for Prometheus zu beginnen. Wählen Sie aus den folgenden Endpunkten mit Servicenamen:

- `com.amazonaws.region.aps-workspaces`

  Wählen Sie diesen Servicenamen, um mit Prometheus-kompatiblen APIs zu arbeiten. Weitere Informationen finden Sie unter Prometheus-kompatible APIs im Amazon Managed Service for Prometheus Benutzerhandbuch.

- `com.amazonaws.region.aps`

  Wählen Sie diesen Servicenamen, um Workspace-Management-Aufgaben auszuführen. Weitere Informationen finden Sie unter Amazon Managed Service for Prometheus APIs im Amazon Managed Service for Prometheus Benutzerhandbuch.

> **Note**
>
> Wenn Sie remote_write in einer VPC ohne direkten Internetzugang verwenden, müssen Sie auch einen VPC-Schnittstellen-Endpunkt für AWS Security Token Service erstellen, damit sigv4 über den Endpunkt arbeiten kann. Weitere Informationen zum Erstellen eines VPC-Endpunktes für AWS STS finden Sie unter AWS STS Schnittstellen-VPC-Endpunkte im AWS Identity and Access Management Benutzerhandbuch. Sie müssen AWS STS festlegen, um regionalisierte Endpunkte zu verwenden.

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 Amazon Managed Service for Prometheus Schnittstelle zu kontrollieren. Weitere Informationen finden Sie im nächsten Abschnitt.

Wenn Sie einen Schnittstellen-VPC-Endpunkt für Amazon Managed Service for 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 for Prometheus verwendet öffentliche Endpunkte oder private Schnittstellenendpunkte (je nachdem, welche verwendet werden), um diese Aufgabe auszuführen.

Steuern des Zugriffs auf Ihren Amazon Managed Service for Prometheus VPC-Endpunkt

Sie können VPC-Endpunktrichtlinien verwenden, um den Zugriff auf Ihren VPC-Endpunkt der Amazon Managed Service for Prometheus-Schnittstelle 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 for Prometheus. Diese Richtlinie ermöglicht Benutzern mit der Rolle `PromUser`, die über die VPC eine Verbindung zu Amazon Managed Service for Prometheus herstellen, Workspaces und Regelgruppen anzuzeigen, aber beispielsweise keine Workspaces zu erstellen oder zu löschen.

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

```
        {
            "Sid": "AmazonManagedPrometheusPermissions",
            "Effect": "Allow",
            "Action": [
                "aps:DescribeWorkspace",
                "aps:DescribeRuleGroupsNamespace",
                "aps:ListRuleGroupsNamespace",
                "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.

```
{
    "Statement": [
        {
            "Action": "aps:*",
            "Effect": "Allow",
            "Principal": "*",
            "Resource": "*",
            "Condition": {
                "IpAddress": {
                    "aws:VpcSourceIp": "192.0.2.123"
                },
        "StringEquals": {
                    "aws:SourceVpc": "vpc-555555555555"
                }
            }
        }
    ]
}
```

Fehlerbehebung

Nutzen Sie die folgenden Abschnitte, um Probleme zu beheben, die Amazon Managed Service für Prometheus aufweist.

Themen

429 Fehler oder Fehler bei Überschreitung des Limits

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 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. Aktive Serien (Standard)

```
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 Amazon Managed Service für Prometheus Service Quotas

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 geht davon aus, 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.

Weitere Informationen darüber, wie das funktioniert oder wie Sie Ihr Setup überprüfen können, finden Sie im Blogbeitrag Understanding Duplicate Samples and O ut-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 Amazon Managed Service für Prometheus Service Quotas.

- Serienlimit pro Benutzer von *<value>* überschritten, bitte wenden Sie sich an den Administrator, um es zu erhöhen
- Serienlimit pro Metrik von *<value>* 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 Amazon Managed Service für Prometheus Service Quotas.

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])`	
M ost-to-least Liste der aktiven Serien pro Metrikname	`sort_desc(count by(__name__)({__name__!=""}))`	

Datentyp	Zu verwendende Abfragen	
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 CloudWatch Metriken.

Grund	Bedeutung
greater_than_max_sample_age	Logzeilen, die älter als die aktuelle Uhrzeit sind, werden gelöscht
new-value-for-timestamp	Doppelte Beispiele werden mit einem anderen Zeitstempel als dem, der zuvor aufgezeichnet wurde, 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	Beispiele werden nicht in der richtigen Reihenfolge versandt und können nicht bearbeitet werden

Grund	Bedeutung
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
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

Markierung

Ein Tag ist eine benutzerdefinierte Attributbezeichnung, die Sie oder AWS einer AWS-Ressource zuweisen. 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 Workspaces bis zu 50 Tags zuweisen.

Tags helfen Ihnen, Ihre AWS-Ressourcen zu identifizieren und zu organisieren. Viele AWS-Services unterstützen das Markieren mit Tags (kurz: Tagging). So können Ressourcen aus verschiedenen Services dasselbe Tag zuweisen, um anzugeben, dass die Ressourcen verbunden sind. Sie können beispielsweise das gleiche Tag eines Amazon Managed Service for Prometheus zuweisen, das Sie einem Amazon S3-Bucket zuweisen. Weitere Informationen zu Markierungsstrategien finden Sie unter Markieren von AWS-Ressourcen.

In Amazon Managed Service for Prometheus können Namensräume sowohl für Workspaces als auch für Regelgruppen mit Tags versehen werden. Sie können die Konsole, AWS CLI, APIs oder SDKs verwenden, um Tags für diese Ressourcen hinzuzufügen, zu verwalten und zu entfernen. Sie können nicht nur Ihre Workspaces und Regelgruppen-Namensräume mit Tags identifizieren, organisieren und verfolgen, sondern auch Tags in IAM-Richtlinien verwenden, um zu kontrollieren, wer Ihre Amazon Managed Service for Prometheus Ressourcen anzeigen und mit ihnen interagieren kann.

Tag (Markierung)-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 Markierungsschema für mehrere AWS-Services und -Ressourcen verwendet wird, denken Sie daran, dass andere Services möglicherweise Einschränkungen für zulässige Zeichen haben. 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. Sie sind für die AWS Verwendung reserviert. Sie können keine Tag-Schlüssel oder -Werte mit diesem Präfix bearbeiten oder löschen. Tags mit diesem Präfix werden nicht als Ihre Tags pro Ressourcenlimit angerechnet.

Themen

Markieren von Workspaces

Verwenden Sie die Verfahren in diesem Abschnitt, um mit Tags für Amazon Managed Service for Prometheus Workspaces zu arbeiten.

Themen

Einem Workspaces ein Tag hinzufügen

Das Hinzufügen von Tags zu einem Amazon Managed Service for Prometheus Workspace 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 mit der Konsole oder AWS CLI Tags zu einem Amazon Managed Service for Prometheus Workspace hinzufügen.

> **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 Amazon Managed Service for Prometheus Workspace während dessen Erstellung finden Sie unter Einen Workspace erstellen.

Themen

Einem Workspace (Konsole) einen Tag hinzufügen

Sie können mit der Konsole einen oder mehrere Tags zu einem Amazon Managed Service for Prometheus Workspace 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 Workspaces, den Sie verwalten möchten.
5. Wählen Sie die Registerkarte Tags aus.

6. Wenn dem Amazon Managed Service for Prometheus Workspace keine Tags hinzugefügt wurden, wählen Sie Tag erstellen. Andernfalls wählen Sie Tags verwalten.
7. Geben Sie für Key (Schlüssel) einen Namen für das Tag ein. Sie können einen optionalen Wert für das Tag unter Value (Wert) hinzufügen.
8. (Optional) Zum Hinzufügen eines weiteren Tags wählen Sie Add tag (Tag hinzufügen) erneut aus.
9. Wenn Sie fertig mit dem Hinzufügen der Tags sind, klicken Sie auf Save changes (Änderungen speichern).

Einem Workspace einen Tag hinzufügen (AWS CLI)

Folgen Sie diesen Schritten und fügen Sie mit AWS CLI einem Amazon Managed Service for Prometheus Workspace einen Tag hinzu. Informationen darüber, wie Sie beim Erstellen einem Workspace einen Tag hinzufügen können, finden Sie unter Einen Workspace erstellen.

Bei diesen Schritten wird davon ausgegangen, dass Sie bereits eine aktuelle Version der AWS CLI installiert oder eine Aktualisierung auf die aktuelle Version vorgenommen 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 Workspaces 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 markieren, einem Tag-Schlüssel namens Status mit dem Tag-Wert Secret und einem Tag-Schlüssel namens Team mit dem Tag-Wert My-Team:*

```
aws amp tag-resource --resource-arn arn:aws:aps:us-
west-2:123456789012:workspace/IDstring
--tags Status=Secret,Team=My-Team
```

Bei erfolgreicher Ausführung gibt dieser Befehl nichts zurück.

Tags für einen Workspace anzeigen

Tags helfen Ihnen, Ihre AWS-Ressourcen zu identifizieren und den Zugriff auf diese zu verwalten. Weitere Informationen zu Markierungsstrategien finden Sie unter Markieren von AWS-Ressourcen.

Anzeigen von Tags für einen Amazon Managed Service for Prometheus Workspace (Konsole)

Sie können die Konsole verwenden, um die Tags anzuzeigen, die einem Amazon Managed Service for Prometheus Workspaces 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 Workspaces, den Sie verwalten möchten.
5. Wählen Sie die Registerkarte Tags aus.

Anzeigen von Tags für einen Amazon Managed Service for Prometheus Workspaces (AWS CLI)

Führen Sie die folgenden Schritte aus, um mit AWS CLI 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 dem des anderen.

> **Important**
>
> Das Bearbeiten von Tags für einen Amazon Managed Service for Prometheus Workspace kann sich auf den Zugriff auf diesen 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 Amazon Managed Service for Prometheus Workspace bearbeiten (Konsole)

Sie können mit der Konsole die Tags bearbeiten, die einem Amazon Managed Service for Prometheus Workspace 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 Workspaces, den Sie verwalten möchten.
5. Wählen Sie die Registerkarte Tags aus.
6. Wenn dem Workspace noch keine Tags hinzugefügt wurden, wählen Sie Add tag (Tag hinzufügen) aus. Andernfalls wählen Sie Tags verwalten.
7. Geben Sie für Key (Schlüssel) einen Namen für das Tag ein. Sie können einen optionalen Wert für das Tag unter Value (Wert) hinzufügen.
8. (Optional) Zum Hinzufügen eines weiteren Tags wählen Sie Add tag (Tag hinzufügen) erneut aus.
9. Wenn Sie fertig mit dem Hinzufügen der Tags sind, klicken Sie auf Save changes (Änderungen speichern).

Bearbeiten von Tags für einen Amazon Managed Service for Prometheus Workspace (AWS CLI)

Führen Sie die folgenden Schritte aus, um über die AWS CLI ein Tag für einen Workspace 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 Amazon Managed Service for Prometheus Workspaces, 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. Das Entfernen eines Tags löscht nicht das Tag anderer AWS-Ressourcen, die mit diesem Tag verknüpft sind.

> **Important**
>
> Das Entfernen von Tags für einen Amazon Managed Service for Prometheus Workspace 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 Amazon Managed Service for Prometheus Workspace (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 Workspaces, 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 Remove (Entfernen) aus.

Ein Tag aus einem Amazon Managed Service for Prometheus Workspace entfernen (AWS CLI)

Führen Sie die folgenden Schritte aus, um über AWS CLI ein Tag aus einem Workspace zu entfernen. Durch das Entfernen wird ein Tag nicht gelöscht, sondern lediglich die Verknüpfung zwischen dem Tag und der Workspace entfernt.

> **Note**
>
> If you delete an Amazon Managed Service for Prometheus workspace, all tag associations are removed from the deleted workspace. 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 Workspaces an, für den Sie Tags hinzufügen möchten, sowie den Tag-Schlüssel des zu entfernenden Tags. So entfernen Sie beispielsweise ein Tag von einem Workspace mit dem Namen My-Workspace und dem Tag-Schlüssel *Status*:

```
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.

Namensräume von Regelgruppen markieren

Verwenden Sie die Verfahren in diesem Abschnitt, um mit Tags für Amazon Managed Service for Prometheus Regelgruppen-Namensräume zu arbeiten.

Themen

Hinzufügen eines Tags zum Regelgruppen-Namensraum

Das Hinzufügen von Tags zu einem Amazon Managed Service for Prometheus mit Regelgruppen-Namensräumen 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 Regelgruppen-Namensraum hinzu. Nach der Erstellung der Tags können Sie IAM-Richtlinien für die Verwaltung des Zugriffs auf den Namensraum basierend auf diesen Tags erstellen. Sie können die Konsole oder AWS CLI verwenden, um Tags zu einem Amazon Managed Service for Prometheus Regelgruppen-Namensraum hinzuzufügen.

> **Important**
>
> Das Hinzufügen von Tags zu einem Regelgruppen-Namensraum kann sich auf den Zugriff auf den Namensraum 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-Namensraum während ihrer Erstellung finden Sie unter Erstellen einer Regeldatei.

Themen

Ein Tag zum Namensraum einer Regelgruppe hinzufügen (Konsole)

Sie können die Konsole verwenden, um einem Amazon Managed Service for Prometheus Regelgruppen-Namensraum 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 Workspaces, den Sie verwalten möchten.
5. Wählen Sie die Registerkarte Regelverwaltung.
6. Klicken Sie auf die Schaltfläche neben dem Namen des Namensraums und wählen Sie Bearbeiten.
7. Wählen Sie Tags erstellen, Neues Tag hinzufügen.
8. Geben Sie für Key (Schlüssel) einen Namen für das Tag ein. Sie können einen optionalen Wert für das Tag unter Value (Wert) hinzufügen.
9. (Optional) Zum Hinzufügen eines weiteren Tags wählen Sie Add new tag (Neues Tag hinzufügen) erneut aus.
10. Wenn Sie fertig mit dem Hinzufügen der Tags sind, klicken Sie auf Save changes (Änderungen speichern).

Fügen Sie dem Namensraum einer Regelgruppe ein Tag hinzu (AWS CLI)

Folgen Sie diesen Schritten und fügen Sie mit AWS CLI einem Amazon Managed Service for Prometheus Regelgruppen-Namensraum einen Tag hinzu. Informationen zum Hinzufügen eines Tags zu einem Regelgruppen-Namensraum, wenn Sie ihn erstellen, finden Sie unter Hochladen einer Regelkonfigurationsdatei auf Amazon Managed Service für Prometheus.

Bei diesen Schritten wird davon ausgegangen, dass Sie bereits eine aktuelle Version der AWS CLI installiert oder eine Aktualisierung auf die aktuelle Version vorgenommen 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-Namensraums an, für die Sie Tags hinzufügen möchten, sowie den Schlüssel und Wert des hinzuzufügenden Tags. Sie können einem Regelgruppen-Namensraum mehrere Tags hinzufügen. Um beispielsweise einen Amazon Managed Service for Prometheus Namensraum namens My-Workspace mit zwei Tags zu markieren, einen Tag-Schlüssel namens *Status* mit dem Tag-Wert *Secret* und einen Tag-Schlüssel namens *Team* mit dem Tag-Wert *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-Namensraum

Tags helfen Ihnen, Ihre AWS-Ressourcen zu identifizieren und den Zugriff auf diese zu verwalten. Weitere Informationen zu Markierungsstrategien finden Sie unter Markieren von AWS-Ressourcen.

Tags für einen Amazon Managed Service for Prometheus Regelgruppen-Namensraum anzeigen (Konsole)

Sie können mit der Konsole die Tags anzeigen, die einem Amazon Managed Service for Prometheus Namensraum 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 Workspaces, den Sie verwalten möchten.
5. Wählen Sie die Registerkarte Regelverwaltung.
6. Wählen Sie den Namensraum „Glue“ aus.

Anzeigen von Tags für einen Amazon Managed Service for Prometheus Workspaces (AWS CLI)

Führen Sie die folgenden Schritte aus, um über AWS CLI die AWS Tags für einen Regelgruppen-Namensraum 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. So zeigen Sie beispielsweise eine Liste von Tag-Schlüsseln und Tag-Werten für einen Regelgruppen-Namensraum 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-Namensraum

Sie können den Wert für ein Tag ändern, das mit einem Regelgruppen-Namensraum 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 dem des anderen.

> **Important**
>
> Das Bearbeiten von Tags für einen Regelgruppen-Namensraum 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 Amazon Managed Service for Prometheus Regelgruppen-Namensraum bearbeiten (Konsole)

Sie können mit der Konsole die Tags bearbeiten, die einem Amazon Managed Service for Prometheus Regelgruppen-Namensraum 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 Workspaces, den Sie verwalten möchten.
5. Wählen Sie die Registerkarte Regelverwaltung.
6. Wählen Sie den Namen des Namensraums 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 Add new tag (Neues Tag hinzufügen) aus.
10. Wenn Sie fertig mit dem Hinzufügen und Bearbeiten der Tags sind, klicken Sie auf Save changes (Änderungen speichern).

Tags für einen Amazon Managed Service for Prometheus Regelgruppen-Namensraum bearbeiten (AWS CLI)

Führen Sie die folgenden Schritte aus, um über AWS CLI ein Tag für einen Regelgruppen-Namensraum 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 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-Namensraum

Sie können ein oder mehrere Tags entfernen, die mit einem Regelgruppen-Namensraum verknüpft sind. Das Entfernen eines Tags löscht nicht das Tag anderer AWS-Ressourcen, 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 Amazon Managed Service für Prometheus Regelgruppen-Namensraum entfernen (Konsole)

Sie können die Konsole verwenden, um die Verknüpfung zwischen einem Tag und einem Regelgruppen-Namensraum 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 Workspaces, den Sie verwalten möchten.
5. Wählen Sie die Registerkarte Regelverwaltung.
6. Wählen Sie den Namen des Namensraums 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 Amazon Managed Service für Prometheus Regelgruppen-Namensraum entfernen (AWS CLI)

Führen Sie die folgenden Schritte aus, um mit AWS CLI ein Tag aus einem Regelgruppen-Namensraum zu entfernen. Durch das Entfernen wird ein Tag nicht gelöscht, sondern lediglich die Verknüpfung zwischen dem Tag und des Regelgruppen-Namensraums entfernt.

> **Note**
>
> Wenn Sie einen Amazon Managed Service for Prometheus Regelgruppen-Namensraum löschen, werden alle Tag-Zuordnungen aus dem gelöschten Namensraum entfernt. Sie müssen keine Tags entfernen, bevor Sie einen Namensraum löschen.

Führen Sie am Terminal oder über die Befehlszeile den Befehl untag-resource aus und geben Sie dabei den Amazon-Ressourcennamen (ARN) des Regelgruppen-Namensraums an, für den Sie Tags hinzufügen möchten, sowie den Tag-Schlüssel des zu entfernenden Tags. So entfernen Sie beispielsweise ein Tag von einem Workspace mit dem Namen My-Workspace und dem Tag-Schlüssel `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.

Amazon Managed Service für Prometheus Service Quotas

In den folgenden beiden Abschnitten werden die mit Amazon Managed Service for Prometheus verbundenen Kontingente und Limits beschrieben.

Service Quotas

Amazon Managed Service for Prometheus hat die folgenden Kontingente. Amazon Managed Service for Prometheus bietet CloudWatch Nutzungsmetriken zur Überwachung der Prometheus-Ressourcennutzung. Mit der Alarmfunktion für CloudWatch Nutzungsmetriken können Sie Prometheus-Ressourcen und -Nutzung überwachen, um Limitfehler zu vermeiden.

Wenn Ihre Projekte und Workspaces wachsen, sind die häufigsten Kontingente, die Sie möglicherweise überwachen oder eine Erhöhung beantragen müssen, folgende Kontingente: Aktive Serien pro Workspace, Aufnahmerate pro Workspace und Aufnahme-Burst-Größe pro Workspace.

Für alle anpassbaren Kontingente kannst du eine Erhöhung des Kontingents beantragen, indem du den Link in der Spalte „Anpassbar“ auswählst, oder indem du eine Erhöhung des Kontingents beantragst.

Das Limit für aktive Serien pro Workspace wird dynamisch angewendet. Weitere Informationen finden Sie unter Aktive Serien (Standard). Die Aufnahmerate pro Workspace und die Aufnahme-Burst-Größe pro Workspace steuern zusammen, wie schnell Sie Daten in Ihren Workspace aufnehmen können. Weitere Informationen finden Sie unter Drosselung der Aufnahme.

> **Note**
>
> Sofern nicht anders angegeben, gelten diese Kontingente pro Workspace.

Name	Standard	Anpas	Beschreibung
Aktive Metriken mit Metadaten pro Workspace	Jede unterstützte Region: 20 000	Nein	Die Anzahl einmaliger aktiver Metriken mit Metadaten pro Workspace.

Name	Standard	Anpas	Beschreibung
Aktive Serien pro Workspace	Jede unterstützte Region: 10.000.000 alle 2 Stunden	Ja	Die Anzahl der einmaligen aktiven Serien pro Workspace. Eine Serie ist aktiv, wenn in den letzten 2 Stunden ein Beispiel gemeldet wurde. Die Kapazität von 2 m bis 10 m wird automatisch an die letzten 30 Minuten der Nutzung angepasst.
Größe der Alert-Aggregationsgruppe in der Alert Manager-Definitionsdatei	Jede unterstützte Region: 1 000	Ja	Die maximale Größe einer Alert-Aggregationsgruppe in der Alert-Manager-Definitionsdatei. Jede Labelwertkombination von group_by würde eine Aggregationsgruppe erstellen.
Größe der Definitionsdatei des Alert Managers	Jede unterstützte Region: 1 Megabyte	Nein	Die maximale Größe einer Alert Manager-Definitionsdatei.
Alert-Nutzlastgröße in Alert Manager	Jede unterstützte Region: 20 Megabyte	Nein	Die maximale Größe der Alert-Nutzlast aller Alert Manager-Warnungen pro Workspace. Die Alert-Größe hängt von den Bezeichnungen und Anmerkungen ab.

Name	Standard	Anpas	Beschreibung
Warnungen in Alert Manager	Jede unterstützte Region: 1 000	Ja	Die maximale Anzahl gleichzeitiger Alert Manager-Warnungen pro Workspace.
HA-Tracker-Cluster	Jede unterstützte Region: 500	Nein	Die maximale Anzahl von Clustern, die der HA-Tracker für aufgenommene Beispiel pro Workspace verfolgt.
Aufnahme-Burst-Größe pro Workspace	Jede unterstützte Region: 1 000 000	Ja	Die maximale Anzahl von Beispielen, die pro Workspace in einem Burst pro Sekunde aufgenommen werden können.
Aufnahme-Rate pro Workspace	Jede unterstützte Region: 170.000	Ja	Metrische Beispiel-Aufnahmerate pro Workspace pro Sekunde.
Sperrregeln in der Alert Manager-Definitionsdatei	Jede unterstützte Region: 100	Yes (Ja)	Die maximale Anzahl von Sperrregeln in der Alert-Manager-Definitionsdatei.
Labelgröße	Jede unterstützte Region: 350 Kilobyte	Nein	Die maximale kombinierte Größe aller Etiketten und Etikettenwerte, die für eine Serie akzeptiert werden.
Labels pro metrischer Serie	Jede unterstützte Region: 70	Ja	Anzahl der Etiketten pro metrischer Serie.

Name	Standard	Anpas	Beschreibung
Länge der Metadaten	Jede unterstützte Region: 1 Kilobyte	Nein	Die maximale Länge, die für metrische Metadaten akzeptiert wird. Metadaten beziehen sich auf Metrikname, HELP und UNIT.
Metadaten pro Metrik	Jede unterstützte Region: 10	Nein	Die Anzahl von Metadaten pro Metrik.
Knoten im Alert Manager-Routingbaum	Jede unterstützte Region: 100	Yes (Ja)	Die maximale Anzahl der Knoten im Alert Manager-Routingbaum.
Anzahl der API-Operationen in Transaktionen pro Sekunde	Jede unterstützte Region: 10	Yes (Ja)	Die maximale Anzahl von API-Vorgängen pro Sekunde pro Region. Dazu gehören Workspace-CRUD-APIs, Tagging-APIs, Regelgruppen-Namensräume-CRUD-APIs und CRUD-APIs für die Alert-Manager-Definition.
Bytes für sofortige Abfragen abfragen	Jede unterstützte Region: 5 Gigabyte	Nein	Die maximale Byte-Anzahl, die mit einer einzigen Sofortabfrage gescannt werden kann.
Byte für Bereichsabfragen abfragen	Jede unterstützte Region: 5 Gigabyte	Nein	Die maximale Byte-Anzahl, die pro 24-Stunden-Intervall in einer einzigen Bereichsabfrage gescannt werden kann.

Name	Standard	Anpas	Beschreibung
Abfrageblöcke abgerufen	Jede unterstützte Region: 20 000 000	Nein	Die maximale Anzahl der Blöcke, die während einer einzigen Abfrage gescannt werden können.
Abfragebeispiele	Jede unterstützte Region: 50 000 000	Nein	Die maximale Anzahl der Beispiele, die während einer einzigen Abfrage gescannt werden können.
Abfrageserie abgerufen	Jede unterstützte Region: 12.000.000	Nein	Die maximale Anzahl der Serien, die während einer einzigen Abfrage gescannt werden können.
Zeitraum für die Abfrage in Tagen	Jede unterstützte Region: 32	Nein	Der maximale Zeitraum einer PromQL-Abfrage.
Anforderungsgröße	Jede unterstützte Region: 1 Megabyte	Nein	Maximale Größe der Abfrage für Aufnahme oder Abfrage.
Aufbewahrungsdauer für erfasste Daten in Tagen	Jede unterstützte Region: 150	Ja	Die Anzahl der Tage, für die Daten in einem Workspace beibehalten werden. Daten, die älter sind, werden gelöscht. Sie können eine Änderung des Kontingents beantragen, um diesen Wert zu erhöhen oder zu verringern.

Name	Standard	Anpas	Beschreibung
Bewertungsintervall für die Regel	Jede unterstützte Region: 30 Sekunden	Ja	Das minimale Regelauswertungsintervall einer Regelgruppe pro Workspace.
Größe der Definitionsdatei des Regelgruppen-Namensraums	Jede unterstützte Region: 1 Megabyte	Nein	Die maximale Größe einer Definitionsdatei des Regelgruppen-Namensraums.
Regeln pro Workspace	Jede unterstützte Region: 2.000	Ja	Die maximale Anzahl von Regeln pro WorkSpace.
Vorlagen in der Alert Manager-Definitionsdatei	Jede unterstützte Region: 100	Yes (Ja)	Die maximale Anzahl der Vorlagen in der Alert Manager-Definitionsdatei.
Workspaces pro Region und Konto	Jede unterstützte Region: 25	Ja	Die maximale Anzahl von Workspaces pro Region.

Aktive Serien (Standard)

Mit Amazon Managed Service for Prometheus können Sie standardmäßig bis zu Ihrem Kontingent aktive Zeitreihen nutzen.

Amazon Managed Service for Prometheus Workspaces passen sich automatisch Ihrem Erfassungsvolumen an. Wenn Ihre Nutzung zunimmt, erhöht Amazon Managed Service for Prometheus automatisch Ihre Zeitreihenkapazität, um Ihre Basisnutzung bis zum Standardkontingent zu verdoppeln. Wenn Ihre durchschnittliche aktive Zeitreihe in den letzten 30 Minuten beispielsweise 3,5 Millionen beträgt, können Sie bis zu 7 Millionen Zeitreihen ohne Drosselung verwenden.

Wenn Sie mehr als das Doppelte Ihres vorherigen Basiswerts benötigen, weist Amazon Managed Service for Prometheus automatisch mehr Kapazität zu, wenn Ihr Aufnahmevolumen steigt, um sicherzustellen, dass Ihre Workload nicht dauerhaft bis zu Ihrem Kontingent gedrosselt wird. Eine Drosselung kann jedoch eintreten, wenn Sie innerhalb von 30 Minuten das Doppelte Ihrer vorherigen Basislinie überschreiten. Um Drosselung zu vermeiden, empfiehlt Amazon Managed Service for

Prometheus, die Aufnahme schrittweise zu erhöhen, wenn sie auf mehr als das Doppelte Ihrer vorherigen aktiven Zeitreihen erhöht wird.

> **Note**
>
> Die Mindestkapazität für aktive Zeitreihen beträgt 2 Millionen. Bei weniger als 2 Millionen Zeitreihen gibt es keine Drosselung.
> Wenn Sie dieses Kontingent überschreiten möchten, können Sie eine Kontingenterhöhung beantragen.

Drosselung der Aufnahme

Amazon Managed Service for Prometheus drosselt die Aufnahme für jeden Workspace, basierend auf Ihren aktuellen Grenzwerten. Dies trägt dazu bei, die Leistung des Workspace aufrechtzuerhalten. Wenn Sie das Limit überschreiten, sehen Sie `DiscardedSamples` in CloudWatch Metriken (mit dem `rate_limited` Grund). Sie können Amazon verwenden, CloudWatch um Ihre Aufnahme zu überwachen und einen Alarm zu erstellen, der Sie warnt, wenn Sie kurz vor dem Erreichen der Drosselungsgrenzen stehen. Weitere Informationen finden Sie unter CloudWatch Metriken.

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 stellt Ihr Aufnahmelimit zu einer bestimmten Sekunde dar.

Jedes aufgenommene Datenbeispiel entfernt ein Token aus dem Bucket. Wenn Ihre Bucket-Größe (Aufnahme-Burst-Größe pro Workspace) 1 000 000 beträgt, kann Ihr Workspace eine Million Datenstichproben pro Sekunde aufnehmen. Wenn es eine Million Stichproben für die Aufnahme überschreitet, wird es gedrosselt und keine weiteren Datensätze aufnehmen. Zusätzliche Datenstichproben werden verworfen.

Der Bucket wird automatisch mit einer festgelegten Rate aufgefüllt. Wenn der Bucket unter seiner maximalen Kapazität liegt, wird ihm jede Sekunde eine festgelegte Anzahl von Tokens wieder hinzugefügt, bis er seine maximale Kapazität erreicht. Wenn der Bucket voll ist, wenn die Auffülltoken eintreffen, werden sie verworfen. Der Bucket darf nicht mehr als seine maximale Anzahl von Token enthalten. Die Nachfüllrate für die Beispielaufnahme wird durch das Limit für die Aufnahmerate pro Workspace festgelegt. Wenn Ihre Aufnahmerate pro Workspace auf 170.000 festgelegt ist, beträgt die Nachfüllrate für den Bucket 170.000 Token pro Sekunde.

Wenn Ihr Workspace 1 000 000 Datenstichproben pro Sekunde aufnimmt, wird Ihr 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. Wenn keine weitere Aufnahme erfolgt, kehrt der zuvor leere Bucket in 6 Sekunden zur maximalen Kapazität zurück.

> **Note**
>
> Die Aufnahme erfolgt in Stapelanforderungen. Wenn Sie 100 Token zur Verfügung haben und eine Anfrage mit 101 Beispielen senden, wird die gesamte Anfrage abgelehnt. Amazon Managed Service for Prometheus akzeptiert Anfragen nicht teilweise. Wenn Sie einen Kollektor schreiben, können Sie Wiederholungen verwalten (mit kleineren Batches oder nach Ablauf einiger Zeit).

Sie müssen nicht warten, bis der Bucket voll ist, bevor Ihr Workspace weitere Datenbeispiele aufnehmen kann. Sie können Token verwenden, wenn sie dem Bucket hinzugefügt werden. Wenn Sie die Nachfülltoken sofort verwenden, erreicht der Bucket seine maximale Kapazität nicht. Wenn Sie beispielsweise den Bucket aufgebraucht haben, können Sie weiterhin 170.000 Datenstichproben pro Sekunde aufnehmen. Der Bucket kann nur dann auf die maximale Kapazität aufgefüllt werden, wenn Sie weniger als 170.000 Datenstichproben pro Sekunde aufnehmen.

Zusätzliche Limits für aufgenommene Daten

Amazon Managed Service for 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.

API-Referenz

In diesem Abschnitt werden die API-Vorgänge und Datenstrukturen aufgeführt, die von Amazon Managed Service for Prometheus unterstützt werden.

Informationen zu diesen API-Vorgängen und ihren Quotas für Serien, Labels und API-Anfragen finden Sie unter Amazon Managed Service for Prometheus Service Quotas im Amazon Managed Service for Prometheus Leitfaden.

Themen

- Amazon Managed Service for Prometheus APIs
- Prometheus-kompatible APIs

Amazon Managed Service for Prometheus APIs

Amazon Managed Service for Prometheus bietet API-Operationen zum Erstellen und Verwalten Ihrer Amazon Managed Service for Prometheus Workspaces. Dazu gehören APIs für Workspaces, Scraper, Alert Manager-Definitionen, Regelgruppen-Namespaces und Protokollierung.

Ausführliche Informationen zu den APIs von Amazon Managed Service for Prometheus finden Sie in der API-Referenz zu Amazon Managed Service for Prometheus.

Verwenden von Amazon Managed Service for Prometheus mit einem - AWS SDK

AWS -Software Development Kits (SDKs) sind für viele gängige Programmiersprachen verfügbar. Jedes SDK bietet eine API, Codebeispiele und Dokumentation, die es Entwicklern erleichtert, AWS Anwendungen in ihrer bevorzugten Sprache zu erstellen. Eine Liste der SDKs und Tools nach Sprache finden Sie unter Tools, auf denen Sie aufbauen AWS können im AWS -Entwicklerzentrum.

> **SDK-Versionen**
>
> Wir empfehlen Ihnen, den neuesten Build des AWS SDK und alle anderen SDKs zu verwenden, die Sie in Ihren Projekten verwenden, und die SDKs auf dem neuesten Stand zu halten. Das AWS SDK stellt Ihnen die neuesten Funktionen und Sicherheitsupdates bereit.

Prometheus-kompatible APIs

Amazon Managed Service for Prometheus unterstützt die folgenden Prometheus-kompatiblen APIs.

Weitere Informationen zur Verwendung von Prometheus-kompatiblen APIs finden Sie unter Abfragen mithilfe von Promethus-kompatiblen APIs.

Themen

CreateAlertManagerAlerts

Der `CreateAlertManagerAlerts` Vorgang erstellt eine Alert 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 eine Alert steht. Im Folgenden wird ein Beispiel für ein Alert-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

Der `DeleteSilence` löscht eine Alert-Stille.

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

Der `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

Der `GetAlertManagerSilence` ruft Informationen über eine Alert-Stille 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 `ListAlerts` Vorgang ruft derzeit aktive Alerts 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

Der `ListAlertManagerAlerts` ruft Informationen über die Alerts ab, die derzeit im Alert Manager im Workspace ausgelöst werden.

Gültige HTTP-Verben:

`GET`

Gültige URI:

`/workspaces/workspaceId/alertmanager/api/v2/alerts`

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 `ListAlertManagerAlertGroups` Vorgang ruft eine Liste von Alert-Gruppen ab, die im Alert Manager im Workspace konfiguriert wurden.

Gültige HTTP-Verben:

`GET`

Gültige URI:

`/workspaces/workspaceId/alertmanager/api/v2/alerts/groups`

URL-Abfrageparameter:

`active` Boolean. Falls wahr, enthält die zurückgegebene Liste aktive Alerts. Der Standardwert ist „true“. Optional

`silenced` Boolean. Wenn der Wert „true“ ist, enthält die zurückgegebene Liste stille Alerts. Der Standardwert ist „true“. Optional

`inhibited` Boolean. Wenn der Wert „true“ ist, enthält die zurückgegebene Liste gesperrte Alerts. Der Standardwert ist „true“. Optional

`filter` Ein Array von Strings. Eine Liste von Matchern, nach denen Alerts gefiltert werden sollen. Optional

`receiver` String. Ein regulärer Ausdruck, der mit Empfängern übereinstimmt, nach denen Alerts 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 `ListAlertManagerSilences` Vorgang ruft Informationen zu den im Workspace konfigurierten Alert-Silences ab.

Gültige HTTP-Verben:

- `GET`

Gültige URI:

- `/workspaces/workspaceId/alertmanager/api/v2/silences`

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` ruft Informationen zu den im Workspace konfigurierten Regeln ab.

Gültige HTTP-Verben:

`GET`

Gültige URI:

`/workspaces/workspaceId/api/v1/rules`

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 `PutAlertManagerSilences` Vorgang erzeugt eine neue Alert Silence oder aktualisiert eine bestehende.

Gültige HTTP-Verben:

`POST`

Gültige URI:

`/workspaces/`*`workspaceId`*`/alertmanager/api/v2/silences`

URL-Abfrageparameter:

`silence`Ein Objekt, das Silence 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.

Duration (Dauer)

A `duration` in einer Prometheus-kompatiblen 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:

None

`RemoteWrite` hat eine Aufnahmerate von 70.000 Proben pro Sekunde und eine Aufnahme-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 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	7. Februar 2024

	einer eigenen Referenz, der Amazon Managed Service for Prometheus API-Referenz, verfügbar. Promethus-kompatible APIs werden weiterhin im Amazon Managed Service for Prometheus User Guide dokumentiert.	
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

AWS-Glossar

Die neueste AWS-Terminologie finden Sie im AWS-Glossar in der AWS-Glossar-Referenz.

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.